STOCHASTIC SEQUENTIAL ASSIGNMENT PROBLEM

BY

GOLSHID BAHARIAN KHOSHKHOU

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2014

Urbana, Illinois

Doctoral Committee:

Associate Professor Xin Chen, Chair
Professor Sheldon H. Jacobson, Director of Research
Assistant Professor Negar Kiyavash
Associate Professor Vinayak Shanbhag, Penn State University

Abstract

The stochastic sequential assignment problem (SSAP), first introduced by [12], studies the allocation of available distinct workers with deterministic values to sequentially-arriving tasks with stochastic parameters so as to maximize the expected total reward obtained from the assignments. The difficulty and challenge in making the assignment decisions is that the assignments are performed in real-time; specifically, pairing a worker with a task is done without knowledge of future task values. This thesis focuses on studying practical variations and extensions of the SSAP, with the goal of eliminating restricting assumptions so that the problem setting converges to that of real-world problems.

The existing SSAP literature considers a risk-neutral objective function, seeking an assignment policy to maximize the expected total reward; however, a risk-neutral objective function is not always desirable for the decision-maker since the probability distribution function (*pdf*) of the total reward might carry a high probability of low values. To take this issue into account, the first part of this dissertation studies the SSAP under a risk-sensitive objective function. Specifically, the assignments are performed so as to minimize the *threshold probability*, which is the probability of the total reward failing to achieve a specified target (threshold). A target-dependent Markov decision process (MDP) is solved, and sufficient conditions for the existence of a deterministic Markov optimal policy are provided. An approximate algorithm is presented, and convergence of the approximate value function to the optimal value function is established under mild conditions.

The second part of this thesis analyzes the limiting behavior of the SSAP as the number of assignments approaches infinity. It is shown in [12] that the optimal assignment policy for the basic SSAP has a threshold structure and involves computing a new set of breakpoints upon the arrival of each task, which is cumbersome for large-scale problems. To address this issue, the second part of this dissertation focuses on obtaining stationary (time-independent) optimal assignment policies that maximize the long-run expected reward per task and are much easier to perform in real-world problems. An exponential convergence rate is established for the convergence of the expected total reward per task to the optimal value as the number of tasks approaches infinity. The limiting behavior of the SSAP is studied in two different settings. The first

setting assumes an independent and identically distributed (IID) sequence of arriving tasks with observable distribution functions, while the second problem considers the case where task distributions are unobservable and belong to a pool of feasible distributions.

The next part of this dissertation basically brings the first two parts together, studying the limiting behavior of the target-dependent SSAP, where the goal is finding an assignment policy that minimizes the probability of the long-run reward per task failing to achieve a given target value. It is proven that the above-mentioned stationary policy (mentioned in the previous paragraph), which achieves the long-run expected reward per task, minimizes the long-run threshold probability in this problem as well. These two objective functions being optimized simultaneously by one assignment policy is interesting, since the threshold criteria, by definition, deviates from the expected total reward criteria; i.e., although it attempts to avoid hitting below a given target level as much as possible, it does not automatically and necessarily guarantee a reasonable performance in terms of the expected reward.

Finally, stochasticity in the SSAP is extended to worker values in the last part of this thesis, where the worker values are assumed to be random variables, taking on new values upon each task arrival. Four models are introduced which analyze this problem from different aspects; e.g., the distribution function of worker values being identical or distinct, the worker values in a given time period being dependent on those in the preceding time periods (or within the same time period), worker values being deterministic but assumed to be functions of time (possibly deteriorating with time), and task values being independent or dependent on each other. For each of these models an optimal assignment policy is presented so as to maximize the expected total reward.

To my mother and father.

Acknowledgments

I would not have been able to finish this journey and get to where I am now, without the help and support of many people. First and foremost, I would like to thank my research advisor, Professor Sheldon Jacobson, for his guidance, kindness, helpful advice, and patience. Apart from teaching me how to become an independent researcher, he has taught me how to be a better person in life. I am deeply grateful for having the opportunity to work with him during my time at the University of Illinois. He always shared his knowledge with me generously, and I always was inspired by his way of thinking and creative ideas. I am forever indebted to him for being extremely supportive of me, believing in me, and reminding me of my strongpoints when research obstacles made it extremely difficult for me to stay optimistic. Without his insightful scientific guidance, moral support, and patience, there is no doubt that I would not be able to complete my work. He has always been extremely understanding and considerate during my time as his humble student, and words fail me to express my thankfulness for this. His kind insightful words made me confident in myself as a person and in my academic work.

I thank Professor Tolga Tezcan for letting me be a part of his team and supervising me for a couple of years, when I first started as a graduate student at the Department of Industrial and Systems Engineering. I am extremely grateful to have had him as my mentor, when I made the critical transition from an undergraduate student to a PhD student. He kindly and patiently guided me, and he never once was frustrated by the process, which I am sure was not such a fast one at first, since I was just a beginner in learning mathematics in depth and doing theoretical work. It is an honor for me, having worked with an extremely knowledgeable, creative, and kind teacher. His patience and support made persistent in my research and helped me not give up.

I am grateful to Professor Xin Chen, for kindly accepting to be a part of my doctoral and preliminary examination committee and for his supervision and insightful suggestions during my first year as a student at the University of Illinois. I admire his intelligent comments on my research and also his remarkable help during my last months as a PhD student. I thank Professor Negar Kiyavash and Professor Uday Shanbhag for serving on my preliminary and final examination committee. I am grateful to Professor Negar Kiyavash for

her creative suggestions on my work, encouragements, and constructive criticism, which immensely helped me get a better idea of where I stand. I am specially thankful to her for never withholding support. I am deeply appreciative of Professor Uday Shanbhag and Professor Angelia Nedich for their patience and guidance, when answering my questions and enlightening me while I attended their lectures. I am indebted to their kindness and support throughout my studies as a graduate student, and most importantly when I was searching for a new research direction. I also would like to express my gratitude to Professor Uday Shanbhag for his ceaseless support during my last months in graduate school and for being an inspiration. I thank Professor Sean Meyn for being an extraordinary teacher and a kindhearted human being. My work has undoubtedly benefited from his intelligent suggestions. I would also like to thank Professor Florin Boca for being a patient teacher and for his help.

I am grateful for various funding sources that supported my studies and research at the University of Illinois. Namely, the Department of Industrial and Systems Engineering and the Department of Computer Science provided fundings for me as a teaching assistant. I would like to thank the Department of Industrial and Systems Engineering for granting me fellowships. This research has been supported in part by the Air Force Office of Scientific Research under Grant No. $FA9550-10-1-0387$ and the National Science Foundation under Grant No. $CMMI-0900226$. The opinions expressed in this dissertation are those of the author and do not necessarily reflect the views of the United States Air Force, National Science Foundation, or the United States Government.

I am extremely thankful to the faculty and staff of the Department of Industrial and Systems Engineering and the Department of Computer Science. I thank Donna Eiskamp, Debi Hilligoss, Holly Kizer, Lynette Lubben, and Andrea Whitesell for their help. My special thanks goes to my labmates Douglas King, Jason Sauppe, David Morrison, Banafsheh Behzad, Estelle Kone, and Arash Khatibi for their unreserved help, company, and good spirits. I am also very thankful to J.D. Robbins and Ruben Proano for their friendship and help.

I feel extremely fortunate and blessed to have known many wonderful friends all over the world, who have enriched my life. I thank Milad Zabihi for his kindness and support. Many thanks goes to my wonderful highschool and college friends Mahsa Derakhshani, Shabnam Alagheh Band, Maryam Saberi, Sara Ghorbani, Maliheh Aramoon, and Sahar Talebi who have been far from me in distance for years, but very close to me in heart. I am grateful to Atousa Soltani for many years of friendship and memories, patience, advice, and ssupport. I thank Farzaneh Rezaee for the memorable moments in college and for her delightful friendship, support, and unforgettable conversations. I am forever indebted to Behtash Babadi for his presence in my life, although we have been apart for about fifteen years. I would not have been able to go through tough times without him believing in me and his heartwarming thoughts. My thanks goes to Milad Rostamkhani

for his sincere friendship and the joyful memories since my childhood. I would like to also thank Sina Khosravi for always cheering me up, for his kind heart, and lightening up my life. My deepest thanks goes to Bahar Rostamkhani for her incredibly beautiful heart and all the lovely memories, which I carry in my heart forever.

I am thankful for the amazing people I have met in Urbana, who have made my journey memorable. I would like to thank Babak Behzad, Kiumars Soltani, Amin Shali, Amir Pahlavan, Reza Vafabakhsh, Zohreh Golshani, Banafsheh Behzad, Sara Behdad, and Sogol Jahanbekam for their good will and friendship. I am grateful to Anita Hund for helping me through the tough times. I extend my thanks to Farzad Yousefian, Amir Saberi, and Arthur Kirkoryan for their friendship and support when I most needed it. I am grateful to Akbar Jaefari for his genuine friendship, hospitality, and joyful memories. I thank Mehdi Saghafi, Nasrin Sarrafi, Maro Aghazarian, and Shiva Razavi for their extreme kindness and support, heartfelt friendship, and all the adventures and memories. My gratitude is extended to Amélie Bernard for her beautiful soul and the unforgettable memories. I would like to thank Vahid Mirshafiee and Maryam Khademiyan for their blissful company, through both the tough and the happy times. I will forever cherish the time we spent together and every single memory. My deepest gratitude goes to Elham Hamed, who has been the most amazing friend through all these years. I am indebted to her for her company and advice, steady and unwavering support, and kind heart. Words cannot express how fortunate I feel to have met her.

I am forever indebted to Reza Hassani for his love and adding colors to my life. I was at my most vulnerable when he stepped into my life, and he has been immensely caring and patient ever since. I am much obliged to him for his presence in my life, gracious heart, and beautiful soul.

Finally, and most importantly, I extend my deepest gratitude to my loving family. I thank my brother, Arya, for having such a beautiful heart. I am genuinely sorry that I could not be there for him, when he needed me the most. I admire his strength, which makes me feel proud of him, and wish him all the happiness in the world. I am extremely grateful to my brother, Soheil, for his thoughtfulness and selfless support. His blissful presence and endless compassion is forever a priceless irreplaceable source of joy in my life. I thank my parents, Roya and Khalil, for their pure boundless love and endless support. I am grateful to them for teaching me that the most important thing in life is to be kind, compassionate, and honest when treating other human beings. There is not a single bit of doubt that I could not get to where I am now, if it were not for their sacrifices and steady faith in me. I am forever indebted to them, and words cannot even begin to express the amount of love and respect I feel for them.

Table of Contents

List of Figures

List of Abbreviations

IID	Independent and Indentically Distributed.
MDP	Markov Decision Process.
pdf	Probability Distribution Function.
pmf	Probability Mass Function.
SSAP	Stochastic Sequential Assignment Problem.

Chapter 1

Introduction

The stochastic sequential assignment problem (SSAP) in its basic form, introduced in [12], addresses the assignment of n IID sequentially-arriving tasks to n available workers (resources), where the random variable X_j denotes the value of the j^{th} task that arrives during time period j, and a fixed (deterministic) value p_i is associated with worker i. The p_i denote the success rates of the workers, where the larger the success rate, the better the worker. Choosing worker i to perform task j renders the i^{th} worker unavailable for future assignments, with the expected reward associated with this assignment given by $p_i x_j$, where x_j is the observed value of the j^{th} task. The objective is to assign the n workers to n arriving tasks so as to maximize the expected total reward obtained from pairing workers with tasks. The difficulty in making the assignment decisions is that the assignments are performed in real-time; specifically, once a task arrives and its value is revealed, it must be paired instantly with one of the available workers, without knowing any of the future task values. It is shown in [12] that there exists numbers

$$-\infty = a_{0,n} \leq a_{1,n} \leq a_{2,n} \leq \cdots \leq a_{n,n} = +\infty, \tag{1.1}$$

such that the optimal choice in the initial stage is to assign the i^{th} best available worker (i.e., the one with the i^{th} highest success rate) if the random variable X_1 falls within the i^{th} highest interval. This process is repeated at each task arrival; e.g., at time period $j = 2$, where $n - 1$ tasks have yet to arrive, a new set of numbers (or equivalently, optimal breakpoints), labeled $\{a_{i,n-1}\}_{i=0}^{n-1}$, is computed. The value of X_2 is then compared against these breakpoints to find the optimal worker for assignment, similar to the case with X_1. The optimal expected total reward obtained from assigning all the n tasks to workers is given by $\sum_{i=1}^{n} p_i a_{i,n+1}$, and hence, $a_{i,n+1}$ is the expected value of the quantity assigned to the worker with the i^{th} smallest value, where n tasks have yet to arrive and $p_1 \leq p_2 \leq \cdots \leq p_n$. Moreover, the $a_{i,n}$ are independent of the worker values and depend on the task values' distribution function. Specifically, these breakpoints are computed recursively from

$$a_{i,n+1} = \int_{a_{i-1,n}}^{a_{i,n}} u \, dG(u) + a_{i-1,n} G(a_{i-1,n}) + a_{i,n} \left(1 - G(a_{i,n})\right), \tag{1.2}$$

for $i = 1, 2, \cdots, n$, with $a_{0,n} = -\infty$ and $a_{n,n} = +\infty$, where G is the cumulative distribution function of task values [12]. The proof technique applies Lemma 1 (due to [16]) to obtain the structure of the optimal policy.

Lemma 1. *(Hardy's Theorem) If $x_1 \leq x_2 \leq ... \leq x_n$ and $y_1 \leq y_2 \leq ... \leq y_n$ are sequences of real numbers, then*

$$max_{(i_1,i_2,...,i_n)\in V} \sum_{j=1}^{n} x_{i_j} y_j = \sum_{j=1}^{n} x_j y_j, \tag{1.3}$$

where V is the set of all permutations of the integers (1,2,...,n).

Hardy's theorem studies a deterministic version of SSAP, where all task values are fixed and observable at the beginning. It implies that the maximum sum is achieved when the smallest of the x's and y's are paired, the second smallest of the x's and y's are paired, and so forth until the largest of the x's and y's are paired.

It is not that much difficult to deduce from [12] that for any $i = 1, 2, \cdots, n-1$, $a_{i,n}$ is the expected value of the task assigned to the worker with the i^{th} smallest value, when $n-1$ tasks have yet to arrive. Equivalently, $\{a_{i,n}\}_{i=1}^{n-1}$ are the expected values of tasks that arrive after the first task and are optimally assigned to the remaining workers during time periods $2, 3, \cdots, n$. Moreover, $a_{i,n}$ is non-decreasing in i, implying that the expected value of the task assigned to the i^{th} best worker is at least as great as that assigned to the $(i+1)^{th}$ best worker, which is intuitive. Analogous to the deterministic version of the SSAP studied in [16], and although we are dealing with a stochastic process of task arrivals in the SSAP as opposed to a set of real numbers with a fixed ranking, the aim is to achieve a ranking of the current and future task values in the SSAP at each time period. To this end, the optimal assignment policy for the SSAP compares X_1, the value of the first task, against $\{a_{i,n}\}_{i=1}^{n}$, the expected values of future tasks, and the first task is assinged to the worker with the same rank; i.e., if task one is ranked as the i^{th} best task by the optimal breakpoints computed at time period $j = 1$, then the optimal decision is to match it with the i^{th} best worker. Such a ranking of task values is obtained upon each task arrival, and the process is repeated until all the n tasks are allocated to the n workers. To clarify the structure of the optimal policy and the way that the interval breakpoints are computed, a small-scale problem is presented in Example 1.

Example 1. Consider a SSAP, where the total number of tasks to arrive is $n = 3$. When the first task arrives with the observed value of x_1, the optimal policy is to assign it to the i^{th} best worker if x_1 falls within the i^{th} highest interval defined by

$$-\infty \qquad a_{1,3} \qquad a_{2,3} \qquad +\infty,$$

where $a_{1,3} := E[X \wedge a_{1,2}]$, $a_{2,3} := E[X \vee a_{1,2}]$, and $a_{1,2} = E[X]$, by (1.2). After the assignment of the first task, the next decision must be made upon the arrival of the second task. Specifically, the optimal policy is

to assign the second task with value $X_2 = x_2$ to the i^{th} best available worker if x_2 falls within the i^{th} highest interval defined by

$$-\infty \qquad a_{1,2} \qquad +\infty.$$

The SSAP has applications in several areas, and various extensions to the problem have been discussed in the literature. For example, [23] studies a variation of the SSAP in aviation security screening systems where sequentially-arriving passengers are assigned to the available screening devices so as to maximize the expected total security of the airport. For passenger screening, a possible measure of security is the number of threat items (any item that can cause harm and damage on an aircract) deteced after screening or the probability of deterring such items from getting past the security checkpoints and onboard a plane. In the SSAP terminology, X_j denotes the assessed threat value (probability of carrying a threat item) for passenger j, and p_i is the probability of detecing a threat item by security device i. Moreover, [31] addresses the problem of allocating sequentially-arriving kidneys to patients on a transplant waiting list in order to maximize the expected value of a clinically-valid measure of transplant success, such as life expectancy. Once a kidney becomes available for transplant, its observed value denotes the kidney type as a function of its quality (size and weight), the donor's age, and the donor's health. Transplant candidates in this setting are viewed as workers in the SSAP terminology, where a worker value is an indicator of patient type; e.g., their age or medical condition. The reward obtained from assigning a kidney to a patient depends both on the kidney and the patient type. As an illustration, kidneys from elderly donor's can be successfully transplated to elderly patients, if the average kidney life is greater than the patient's remaining life expectancy. Another application of the SSAP is the asset selling problem [5], where one needs to choose the best offers out of a sequence of bids from potential buyers to maximize the expected profit; e.g., a house seller has to decide which bid on the house to accept out of a sequence of sequentially-observed bids.

This thesis focuses on studying practical variations and extensions of the SSAP, with the goal of eliminating restricting assumptions so that the problem setting converges to that of real-world problems. Throughout this work, the number of tasks is assumed to be equal to the number of workers. To relax this assumption, let m denote the number of tasks, while n is the number of workers. If $m > n$, then we add $m - n$ phantom workers with success rates of zero, while if $m < n$, the $n - m$ workers with the smallest values are dropped so that only those m workers with the highest success rates can be chosen. With such a modification, the number of tasks equals the number of workers, and hence, the problem is simplified to the n-task, n-worker model. Section 1.1 elaborates on the motivation behind this dissertation and provides a brief discussion on the results of each chapter.

1.1 Motivation and Contributions

Four different models and problems in the SSAP setting, each discussed in a separate chapter, are studied in this dissertation. In Chapter 2, risk-sensitivity is introduced into the problem by performing the assignments under a risk-sensitive objective function. The existing SSAP literature considers a risk-neutral objective function, seeking an assignment policy to maximize the expected total reward; however, a risk-neutral objective function is not always desirable for the decision-maker since the probability distribution function (*pdf*) of the total reward might carry a high probability of low values, and there are instances that a decision-maker is interested in a stable reward level. To take this issue into consideration, Chapter 2 studies the SSAP where the assignments are performed to minimize the *threshold probability*, which is the probability (risk) of the total reward failing to achieve a specified target (threshold). Specifically, let

$$R_n^{\phi} = \sum_{i=1}^{n} p_{\phi(\tilde{d}_i)} X_i,$$

denote the total reward obtained after assigning all n tasks to available workers under policy ϕ, where $\phi(\tilde{d}_i)$ is the index of the worker assigned to the i^{th} task under ϕ. For a given target value τ, the goal is to find an optimal policy ϕ^* that achieves the following infimum:

$$\inf_{\phi \in \Phi} P_{\phi} \left\{ R_n^{\phi} \leq \tau \right\},$$

where Φ is the set of all admissible policies. For simplicity, the target-dependent stochastic sequential assignment problem is denoted here as the TSSAP in Chapter 2. As an illustration, consider a SSAP which assigns sequentially-arriving passengers to available aviation security resources as they check in at an airport. It is of extreme importance to obtain a stable level of security at an airport. Note that although this security level might not necessarily be the highest possible, a critical goal is to maintain a reasonable security level at all times. In other words, the decision-maker needs to assure that the total security is at least as great as a specified value with high probability, which calls for performing passenger assignments to security devices under the threshold criteria. Specifically, the time interval for screening passengers is divided into n slots (time periods), where passenger j arrives during time period j. Upon the arrival of each passenger, a pre-screening system determines their threat value and classifies them as non-selectees (i.e., the passengers who have been cleared of posing a risk) or selectees (i.e., those who have not been cleared, based on available information known about them [19]). Each assessed threat value denotes the probability that a passenger carries a threat item, with X_j indicating the threat value of passenger j. The capacity of the selectee class (i.e., the number of available screening devices associated with the selectee class) is c, and n denotes the capacity of the non-selectee class. The security level is defined to be the conditional probability of detecting

a passenger with a threat item given that they are classified as selectees or non-selectees. Let L_S and L_{NS} be the security levels associated with the selectee and non-selectee classes, and let $\gamma_j = 1$ and $\gamma_j = 0$ denote the j^{th} passenger assignment as a selectee and a non-selectee, respectively. The *total security* for this setting is defined as

$$\sum_{j=1}^{n} X_j \left[L_S \gamma_j + L_{NS}(1 - \gamma_j) \right],$$

where the objective is to find a policy for assigning passengers to security classes as they check in so as to minimize the probability of the total security failing to achieve the target τ.

Chapter 2 tackles the problem in two settings. Initially, IID task values following a probability mass function (*pmf*) of countable support are considered, and the problem is modelled as countable-state-space target-dependent Markov decision process (MDP). Optimality equations are presented, which achieve a deterministic Markov optimal assignment policy. In addition, an algorithm (proposed originally in Boda) to approximate the optimal value function and the optimal policy is studied, with its useful properties presented in Chapter 2. Although it is shown with numerical examples by [11] that this algorithm approximates the optimal value function extremely well, while reducing the computation time dramatically, the results in Chapter 2 indicate that the gap between the optimal and the approximate value function has a fixed lower bound, resulting from the jump point discontinuities of the optimal value function. These discontinuities are in fact the direct consequence of the assumption that task values are discrete random variables, having a countable *pmf* support. To enhance the performance of this approximate algorithm, the TSSAP is generalized in the second setting to the case where task values are continuous random variables. The target-dependent MDP, which models the problem in the second setting, has an uncountable state space, while the countability of state space is a basic assumption in the existing target-dependent MDP literature (see [33], [11], and [26]). Sufficient conditions for the existence of a deterministic Markov optimal policy, optimality equations, and fundamental characteristics of the optimal value function and the optimal policy are presented for this setting. Specifically, it is proven that once the assumption of countable support for task values is dropped, the optimal value function no longer contains jump point discontinuities. In fact, Lipschitz continuity of the optimal value function is proven for this case. Afterwards, the above-mentioned approximate algorithm is adapted to the uncountable-state-space model, and convergence of the approximate value function to the optimal value function is established under certain mild conditions. In other words, the gaps with fixed lower bounds, that undermine the performance of the approximate algorithm, no longer exist in this setting.

As described in [12], the optimal assignment policy for the SSAP has a threshold structure, and its implementation involves calculating a new set of breakpoints upon the arrival of each task. Specifically, whenever n tasks have yet to arrive, a set of $n-1$ breakpoints are computed, which characterize the optimal

policy and divide the real line into n disjoint intervals (see (1.1)). In other words, the number of breakpoints increase with the problem size (i.e., the number of tasks to arrive), and computations must be performed upon each task arrival. Obtaining breakpoint values is cumbersome for large scale problems such as aviation screening, where passengers are assigned to aviation security resources based on their perceived threat levels to optimize airport security [23]. Assigning passengers to security devices by re-calculating the breakpoints, every time a passenger arrives, is not practical. In this light, Chapter 3 focuses on the limiting behavior of the $\{a_{i,n}\}$ and finding simpler solutions to the SSAP, which are implementable in real-world problems, as n approaches infinity. Specifically, consider the SSAP with n tasks and k (fixed) worker categories, where the i^{th} worker-category consists of r_i workers each with value p_i. Let [.] denote the floor function where $[y] := \max\{m \in \mathbb{Z} | m \leq y\}$. Moreover, let π_i be the fraction of total number of workers that belong to categories $i+1$ to k for $i = 0, 1, 2, \cdots, k-1$, and hence, $\alpha_i := \pi_{i-1} - \pi_i$ denotes the fraction of workers assigned to class i. For simplicity, the i^{th} worker category is referred to as the type-i workers, and the size of the i^{th} category is given by

$$r_i = [n\pi_{i-1}] - [n\pi_i] \quad \text{for } i = 1, 2, \cdots, k,$$

where $\pi_0 = 1$, $\pi_k = 0$, and $\pi_{i+1} < \pi_i$ for $i = 0, 1, \cdots, k-1$. Also, assume that $p_{i+1} < p_i$ for $i = 1, 2, \cdots, k$. The goal is to come up with an optimal assignment policy that achieves the maximum long-run expected reward per task and at the same time is more straightforward structure-wise as that proposed in [12]. Chapter 3 analyzes two variations of this problem. The first setting assumes that tasks are IID with an observable distribution function, while the second problem considers the case that task values are random variables coming from r different distributions, where the successive distributions are governed by an ergodic Markov chain. To be more specific, once a task arrives in a given time period, its value is observed; however, its underlying distribution is unobservable and can actually be any of the above-mentioned r distribution functions. In fact, the second problem deals with the more realistic environment of incomplete information, since the assumption that the underlying distribution of the arriving tasks is known apriori and observable does not hold in most real-world problems. Chapter 3 presents simple stationary policies for both problems, which achieve the optimal long-run expected reward per task. These policies are characterized by $k-1$ fixed time-independent breakpoints, as opposed to the policy described by [12] in which the number of breakpoints increases with n and the time-dependent breakpoints are recalculated each time a task arrives. Furthermore, convergence rate of the expected reward per task to the optimal value under this stationary policy is obtained for both problems. The idea behind these stationary assignment policies is extrapolated from Hardy's theorem [16], which studies the deterministic equivalent of the SSAP. In short, this theorem states that in order to achieve the maximum sum, when pairing the elements of two finite sequences of real numbers, the elements with the i^{th} highest values in each sequence much be paired. To emulate this concept

in the stochastic environment of infinite number of tasks, it is first observed that k worker classes (types) exist, where class i (having the i^{th} highest value among all workers) contains $\%\alpha_i$ of the total number of workers. Accordingly, k hypothetical task categories (classes) are formed, by carefully dividing the domain of task value distributions, such that type-i tasks have the i^{th} overall values and occur with probability α_i. Afterwards and analogous to Hardy's theorem, a type-i task is paired with a type-i worker upon arrival.

Although the main result of Chapter 3, which is the existence of a stationary optimal assignment policy solving the large scale SSAP, seems useful in real world, the same concern (regarding the risk-neutral expected value objective function) hols for this result as well. Particularly, one can still dispute the efficacy of this optimal policy when implemented in a real-world setting, as the *pdf* of the total reward per task might carry high probabilities of low undesirable values. This issue has motivated studying the limiting behavior of the target-dependent SSAP in Chapter 4, which connects the results of its preceding two chapters. The problem setting in Chapter 4 is analogous to that of that in Chapter 3; specifically, an IID sequence of arriving tasks with an observable distribution function is considered first, and the results are then extended to the case with unobservable distributions. Similar to Chapter 3, k worker categories make up the model in Chapter 4, with the i^{th} category consisting of r_i workers of value p_i such that $\sum_{i=1}^{k} r_i = n$. The goal is to find a policy that minimizes the probability of the long-run reward per task failing to achieve a given target τ. This is mathematically expressed as

$$\inf_{\phi\in\Phi} P\left\{\limsup_{n\to+\infty} \frac{1}{n} R_n^{\phi} \leq \tau\right\}, \tag{1.4}$$

where Φ is the set of all admissible policies. The analysis in Chapter 3 indicates that ϕ^* (the optimal policy presented in Chapter 3) achieves the infimum in (1.4). In other words, a stationary assignment policy exists that simultaneously achieves the maximum long-run expected reward per task and minimizes the long-run threshold probability. In fact, this result relieves the decision-maker of the above-mentioned concerns regarding the inadequacy of this risk-neutral class of objective functions.

Chapter 5 studies the SSAP from an aspect different than the previous chapters. The existing SSAP literature assumes the worker values to be deterministic and known in advance, and hence, all the stochasticity in the problem rises from the sequence of arriving tasks with stochastic parameters. Chapter 5 considers an extension of SSAP, in which the success rates are random variables or time-dependent functions, taking on new values upon each task arrival, where the objective function is to maximize the expected total reward. For example, consider the asset selling problem, where a sequence of bids (tasks) arrive from potential buyers, and a set of items (workers) are available to be sold to these buyers so as to maximize the expected total profit. A time-dependent random price is associated with each item. Each item price is dependent on the economic conditions upon the arrival of the offer and hence takes on a new value upon each task arrival, since economic conditions vary from time to time. This can be modelled as a set of distribution functions,

governed by a Markov chain, which generate item prices. Markov chain transitions represent the variations in economic conditions (equivalently, switching from one price distribution to another). Specifically, let Z_j denote the state of the Markov chain with state space $\mathcal{S} = \{1, 2, \cdots, r\}$ at time period j, then $Z_j = k$ implies that item prices upon the arrival of offer j are derived from distribution function F_k, which corresponds to the economic conditions at time j. Another example is an application of SSAP in kidney transplant problems, where a sequence of kidneys become available to patients on a transplant waiting list. The sequence of kidneys is treated as the stream of arriving tasks, whereas patients are treated as workers in this scenario, according to SSAP terminology. Organs available for transplant are scarce resources, and hence, the inter-arrival times between any two such organs is noticeable (i.e., organs do not arrive instantly one after the other). The wait-listed patients, thus, experience a decay in their health conditions while waiting, which leads to modelling patient types (worker values) as deterministic function of time or as random variables in this setting.

For a problem of size n, let $P_k = (p_k(1), p_k(2), \cdots, p_k(n-k+1))$ be the vector of success rates at time period k, with $p_k(i)$ being the success rate of the i^{th} worker at that time. Various versions of the SSAP are studied in Chapter 5, where worker success rates are no longer modelled as fixed deterministic values, and closed form expressions for an optimal assignment policy and optimal expected total reward is provided for each version. Four classes of problems are discussed, analysing different aspects of the SSAP. Class I, which is made up of two models, assumes that for an arbitrarily fixed time period k, $\{p_k(i)\}_{i=1}^{n-k+1}$ is an IID sequence of random variables, where the success rates are independent of task values throughout the problem. Moreover, the vector of success rates at time period k is independent of $\{P_i\}_{i=1}^{k-1}$. Task values are not assumed to be IID, as any kind of dependency is allowed between them and they are not bound to follow the same distribution. Class II discusses three models while modifying and relaxing the independence assumption, made in Class I models, between worker values (either at a given time period or within different time periods); e.g., $\{p_k(i)\}_{k=1}^{n}$ is allowed to form a sequence of dependent random variables. Furthermore, whenever the worker values are considered to have distinct distributions at a given time period, a fixed ranking is assumed for them throughout the assignments.

Class I and Class II models hypothesize that task and worker values are independent of one another, but this assumption is relaxed in Class III. As mentioned above, Class II models relax the assumption of identically-distributed worker values at a given time period; however, in order to deal with this, a restriction is imposed on the model, which is considering a fixed ranking of worker values throughtout the assignments. Classes III and IV primarily deal with eliminating this restriction. Two submodels are analyzed in Class III, where worker values are deterministic functions of time in the first model and random variables taking on new vlues at each task arrival in the second model. Workers belong to two distinct categories, where worker

values are equal within a category. One of the categories is the set of workers all having success rates of zero, whereas the other category is made up of workers with equal but non-zero success rates. Class IV, which consists of two models itself, is to some extent a generalization of its preceding class and allows workers to form two or three distinct classes all with non-zero success rates. Worker values in different categories at a given time period are derived from distinct distribution functions. Most importantly, no assumption is made about the ranking of worker values; i.e., no information is available regarding their order, and their ranking can change from one time period to another (as success rates take on new values upon each task arrival). For example, for a problem of size $n = 3$ with three worker categories, these models characterize a closed-form expression for the optimal assignment policy, where worker values at time period j are denoted by $\alpha_j \sim f_\alpha$, $\beta_j \sim f_\beta$, and $\gamma_j \sim f_\gamma$, where f_α, f_β, and f_γ are distinct *pdf*'s. The procedure to obtain optimal policies for similar problems of larger size is briefly discussed at the end of this chapter.

Finally, Appendix A provides detailed proofs for a few theorems presented in Chapter 3. Appendix B is dedicated to some unpublished results on the SSAP with random number of tasks, where the IID assumption between the task values is relaxed.

Chapter 2

Stochastic Sequential Assignment Problem With Threshold Criteria

2.1 Introduction

Consider the stochastic sequential assignment problem (SSAP) introduced by [12]: There are n workers available to perform n IID sequentially-arriving tasks, where the random variable X_j denotes the value of the j^{th} task that arrives during time period j, and a fixed value (or success rate) p_i is associated with worker i. Whenever the i^{th} worker is assigned to the j^{th} task, the worker becomes unavailable for future assignments, with the expected reward associated with this assignment given by $p_i x_j$, where x_j is the observed value of the j^{th} task.

Several extensions to the stochastic sequential assignment problem have been discussed in the literature. For example, [2], [3], and [28] study the SSAP with various task-arrival-time distributions. Moreover, [22] considers a variation of SSAP in which the number of tasks is unknown until after the final arrival and follows a given probability distribution. An application of SSAP in kidney allocation to patients is addressed by [31], while [19] and [23] address applications of SSAP in aviation security. Existing SSAP literature focuses on a risk-neutral objective function, seeking a policy that maximizes the expected total reward obtained from the sequential assignment of tasks to workers. However, a risk-neutral policy is not always desirable since the probability distribution function (*pdf*) of the total reward may carry with it a high probability of low unaccepted values; therefore, there are instances that a decision-maker is interested in a stable reward and looks for a risk-sensitive optimal assignment policy.

The work presented here is distinct from the existing literature in two ways. First, it considers the SSAP under a different objective function, termed the *threshold criterion*, which seeks to find a policy that minimizes the *threshold probability*: the probability (or risk) of the total reward failing to achieve a specified value (target or *threshold*). Specifically, let

$$R_n^{\phi} = \sum_{i=1}^{n} p_{\phi(\tilde{d}_i)} X_i,$$

denote the total reward obtained after assigning all n tasks to available workers under policy ϕ, where $\phi(\tilde{d}_i)$ is the index of the worker assigned to the i^{th} task under ϕ. For a given target value τ, the goal is to find an

optimal policy ϕ^* that achieves the following infimum:

$$\inf_{\phi \in \Phi} P_\phi \left\{ R_n^\phi \leq \tau \right\},$$

where Φ is the set of all admissible policies. For simplicity, the target-dependent stochastic sequential assignment problem is denoted here as the TSSAP, and a n-stage TSSAP refers to a TSSAP with n tasks and n workers.

The second distinction between the work presented here and the existing literature is that the problem is modelled as a Markov decision process (MDP) and results in an uncountable-state-space MDP, while the countability of the state space is a basic assumption in the existing target-dependent, risk measure literature. Hence, the present work extends the threshold criteria literature to uncountable-state-space MDP's and obtains sufficient conditions for the existence of a deterministic Markov optimal policy. Fundamental characteristics of the optimal value function and the optimal policy are also presented. Finally, the algorithm proposed by [11] to approximate the optimal value function is adapted to the uncountable-state-space model, and convergence of the approximate value function to the optimal value function is established under certain conditions.

Several authors have studied Markov decision processes with the threshold criterion. As mentioned before, the focus of these papers is on Markov decision processes over a countable state space. A finite state space MDP with a bounded reward set is considered in [32], and the optimal value function is characterized by an optimality equation. It is shown in [33] that the optimal value function is a distribution function of the target value and proves the existence of an optimal deterministic Markov policy. Sufficient conditions for the existence of an optimal policy for an infinite horizon MDP over a countable state space are provided by [26]. An algorithm is proposed by [11] to approximate the optimal value function, which decreases the computation time significantly. Moreover, [29] considers undiscounted semi-Markov decision processes with countable state and action spaces, with the objective of minimizing the threshold probability. The existence of an optimal stationary policy is proven, and value iteration methods and a policy improvement method are proposed. Other variations and applications of the threshold problem are discussed by [24], [25], and [27].

This chapter is organized as follows. Section 2.2 mentions potential examples and applications of the TSSAP. Section 2.3 studies the model of a n-stage TSSAP with discrete task values, describes it as a MDP with a countable state space, and presents optimality equations so as to find a policy that minimizes the threshold probability. Section 2.4 discusses exact and approximate methods to solve the optimality equations given in Section 2.3. Section 2.5 extends the model of a n-stage TSSAP to the case where the *pdf* of task values has uncountable support, which results in a MDP with an uncountable state space. Furthermore,

sufficient conditions for the existence of an optimal policy under the threshold criterion and optimality equations to derive the optimal policy are presented. The approximate algorithm discussed in Section 2.4 is adapted to the generalized TSSAP, and its behavior is studied. Section 2.6 presents numerical results, and finally, Section 2.7 provides concluding comments and future directions of research.

2.2 Illustrative Examples

This section provides an example for the TSSAP which demonstrates the application of the threshold criteria. Consider a SSAP which allows sequentially-arriving passengers to be assigned to available aviation security resources as they check in at an airport. The time interval for screening passengers is divided into n slots (stages), where passenger j arrives during stage j. Upon the arrival of each passenger, a pre-screening system determines their threat (risk) value, classifying them as non-selectees (i.e., the passengers who have been cleared of posing a risk) or selectees (i.e., those who have not been cleared, based on available information known about them [19]). Each assessed threat value is defined as the probability that a passenger carries a threat item, and the *pdf* for passengers' threat values is denoted by f, with X_j indicating the threat value of passenger j. The capacity of the selectee class (i.e., the number of available screening devices associated with the selectee class) is c, and n denotes the capacity of the non-selectee class. Define the security level to be the conditional probability of detecting a passenger with a threat item given that they are classified as selectees or non-selectees, and let L_S and L_{NS} be the security levels associated with the selectee and non-selectee classes. Moreover, let $\gamma_j = 1$ and $\gamma_j = 0$ denote the j^{th} passenger assignment as a selectee and a non-selectee, respectively. The *total security* for this setting is defined as

$$\sum_{j=1}^{n} X_j \left[L_S \gamma_j + L_{NS}(1 - \gamma_j) \right],$$

where the objective is to find a policy for assigning passengers to classes as they check in so as to minimize the probability of the total security failing to achieve the target τ.

In the airport security problem the decision-maker needs to make sure that the total reward obtained is at least as great as a specified value with high probability. In other words, it is critical to obtain a stable level of security at all times. Note that although this security level might not be necessarily the highest possible, a critical goal is to maintain a reasonable security level at all times. Section 2.3 studies the n-stage TSSAP with discrete task values, describes it as a MDP, and presents optimality equations so as to find a policy that minimizes the threshold probability.

2.3 Model Description

Consider the original SSAP introduced by [12] where n workers are available to perform n IID sequentially-arriving tasks. A random variable X_j denotes the value of the j^{th} task that arrives during time period j, with a fixed value (or success rate) p_i associated with worker i. If the i^{th} worker is assigned to the j^{th} task with observed value x_j, the worker becomes unavailable for future assignments, and the expected reward due to this assignment is given by $p_i x_j$. Throughout this subsection, it is assumed that the number of tasks equals the number of workers. To relax this assumption, let m denote the number of tasks, while n is the number of workers. If $m > n$, then we add $m - n$ phantom workers with success rates of 0, while if $m < n$, the $n - m$ workers with the smallest values are dropped so that only those m workers with the highest success rates can be chosen. With such a modification, the number of tasks equals the number of workers, and hence, the problem is simplified to the n-task, n-worker model. Unlike the existing SSAP literature, the problem studied here is under an objective function other than maximizing the expected total reward; specifically, the goal is to find a policy that minimizes the probability (or risk) of the total reward failing to achieve a target value, after assigning all the sequentially-arriving tasks to available workers.

Consider the n-stage TSSAP and assume that the IID sequentially-arriving tasks take on values in the set $\mathcal{S} \subseteq [0, +\infty)$; in addition, a vector $P = (p_1, p_2, \cdots, p_n)$ is given with p_i denoting the success rate (or value) of the i^{th} worker, where worker values are considered to be strictly positive. Given P and for $k = 1, 2, \cdots, n$, let

$$\mathcal{W}_k^P := \left\{ (q_1, q_2, \cdots, q_n) \,\middle|\, q_i \in \{0, p_i\} \text{ for } i = 1, 2, \cdots, n \text{ such that } \sum_{j=1}^{n} I_{\{q_j \neq 0\}} = n - k + 1 \right\}$$

denote the set of all possible vectors of worker values at time period k before the assignment of the k^{th} task. Any element of the set $\mathcal{W}_k^P$ has $k - 1$ zero entries, which correspond to the workers that are no longer available, since they have been assigned to previous tasks over the first $k - 1$ time periods. Note that by definition, $\mathcal{W}_1^P = \{P\}$.

Let $\tilde{s}_k$, a_k, and r_k denote the state of the system, the action taken by the decision-maker, and the reward obtained at time period k, respectively. Then, the state of the system at stage k is defined by

$$\tilde{s}_k = (x_k, P^{(k)}) \in \tilde{\mathcal{S}}_k := \mathcal{S} \times \mathcal{W}_k^P,$$

where x_k and $P^{(k)}$ indicate the observed value of the k^{th} task and the vector of success rates at time period k upon the arrival of the k^{th} task, respectively. The state space of the system is thus defined by $\tilde{\mathcal{S}} := \cup_{k=1}^{n} \tilde{\mathcal{S}}_k$. In this section, assume that the state space of task values $\mathcal{S}$ is countable (i.e., $\{X_j\}$ are discrete random variables); the n-stage TSSAP where the task values are continuous random variables with an uncountable state space $\mathcal{S} \subseteq [0, +\infty)$ is studied in Section 2.5. The objective of the TSSAP signifies the decision-maker's need to consider the target level along with the original state of the system at each decision instance, and hence, the state space of the MDP must be enlarged so as to incorporate the target level at each time period.

To this end, define $\tilde{\mathcal{D}} := \cup_{k=1}^{n} \tilde{\mathcal{D}}_k$ to be the updated state space of the MDP (referred to as the state space of the decision-maker) where $\tilde{\mathcal{D}}_k := \tilde{\mathcal{S}}_k \times \mathbb{R}$. Note that $\tilde{\mathcal{S}}$ is the state space of the system and should not be confused with $\tilde{\mathcal{D}}$. At time period k, the action space A at each state is given by the set of workers available for assignment at that state, and hence, $A(x_k, P^{(k)}) = \left\{ i \middle| P_i^{(k)} \neq 0, 1 \leq i \leq n \right\}$ where $P_i^{(k)}$ denotes the i^{th} element of $P^{(k)}$, and $\mathcal{A} := \cup_{s \in \tilde{\mathcal{S}}} A(s)$ is the overall action space. It is obvious from this definition that $A(x_k, P^{(k)})$ is independent of x_k; therefore, the action space at a given state $(x_k, P^{(k)})$ can also be denoted by $A(P^{(k)})$. If at time period k and upon the arrival of the k^{th} task, x_k, action $a_k = i \in A(P^{(k)})$ is chosen, then the target level is decreased by the realized reward amount, $r_k = p_i x_k$. Given that the value of the task at time period $k+1$ is x, the conditional transition probability corresponding to this state change is defined by

$$f(x) = P\left\{ \tilde{d}_{k+1} = ((x, P^{(k)} - p_i e_i), t - p_i x_k) \middle| \tilde{d}_k = ((x_k, P^{(k)}), t), a_k = i \right\},$$

for $\tilde{d}_k \in \tilde{\mathcal{D}}_k$, $\tilde{d}_{k+1} \in \tilde{\mathcal{D}}_{k+1}$, and $i \in A(P^{(k)})$, and $k = 1, 2, \cdots, n-1$, where f is the underlying probability mass function (pmf) of task values with support $\mathcal{S}$.

Let H_k denote the set of all admissible histories up to time period k. Given H_k, a *decision rule* ϕ_k at time period k is a conditional probability measure on the action space $\mathcal{A}$ such that

$$\phi_k(A(P^{(k)}) \mid h_k) = 1,$$

for all $h_k \in H_k$ and $k = 1, 2, \cdots, n$. A decision rule ϕ_k, which is applied at time period k upon the arrival of the k^{th} task, is *deterministic* if it is a mapping from H_k onto $\mathcal{A}$ (i.e., $\phi_k(h_k) \in A(P^{(k)})$ for any $h_k \in H_k$). Consider ϕ_k, an arbitrary deterministic decision rule at time period k. ϕ_k is called a *continuous* decision rule in the target value over the interval $[0, \tau] \subset \mathbb{R}$ if for each $((x, P^{(k)}), t) \in \tilde{\mathcal{S}}_k \times [0, \tau]$, there exists $\epsilon > 0$ such that $\phi_k((x, P^{(k)}), s) = \phi_k((x, P^{(k)}), t)$, for all $s \in (t - \epsilon, t + \epsilon)$. Moreover, a sequence $\phi = (\phi_n, \phi_{n-1}, \cdots, \phi_1)$ of decision rules is called a *policy* for a n-stage TSSAP. If ϕ_k only depends on the current state at time k for all $k = 1, 2, \cdots, n$, then the policy ϕ is a *Markov* policy. In addition, ϕ is called a *deterministic* policy if ϕ_k is deterministic for all $k = 1, 2, \cdots, n$. Let Φ, Φ_D, Φ_M, and Φ_{DM} denote the sets of all policies, all deterministic policies, all Markov policies, and all deterministic Markov policies, respectively. Finally, define

$$\phi \mid_l := (\phi_n, \phi_{n-1}, \cdots, \phi_{n-l+1}),$$

for any $1 \leq l \leq n$.

Fix an arbitrary policy $\phi \in \Phi$, and define the target-dependent risk measure over the last k time periods under $\phi \mid_k$ as

$$V_k^{\phi}((x, P^{(n-k+1)}), t) := P_{\phi}\left\{ R_k \leq t \middle| \tilde{d}_{n-k+1} = ((x, P^{(n-k+1)}), t) \right\},$$

for all $\tilde{d}_{n-k+1} \in \tilde{\mathcal{D}}_{n-k+1}$, where $\tilde{d}_{n-k+1}$ denotes the state of the decision-maker at time period $n-k+1$ (before the assignment of the $(n-k+1)^{th}$ task) and the superscript ϕ in R_k^{ϕ} is dropped to simplify the

notation. Therefore, the optimal value function over n time periods is given by

$$V_n^{\phi^*}((x,P),t) = \inf_{\phi\in\Phi} P_\phi\left\{R_n \le t \middle| \tilde{d}_1 = ((x,P),t)\right\}$$
$$= \inf_{\phi\in\Phi} V_n^{\phi}((x,P),t),$$

for all $\tilde{d}_1 \in \tilde{\mathcal{D}}_1$. Observe that for each $\phi \in \Phi$, $V_n^{\phi}((x,P),t) = 0$ if $t < 0$ since task values are assumed to be non-negative, and hence, $V_n^{\phi^*}((x,P),t) = 0$ for all $t < 0$.

For the n-stage TSSAP described here, the state space of the system $\tilde{\mathcal{S}}$ is countable; in addition, the action space $\mathcal{A}$ is finite. Therefore, Theorem 1 in [33] implies that an optimal policy exists and is in fact deterministic Markovian (i.e., $\phi^* \in \Phi_{DM}$); moreover, the following recursive optimality equations are used to derive the optimal policy and the minimum risk of failing to achieve the target value in the n-stage TSSAP:

$$V_1^{\phi^*}((x,P^{(n)}),t) = I_{\{t\ge p_i x\}}, \tag{2.1}$$

for all $((x,P^{(n)}),t) \in \tilde{\mathcal{D}}_n$ where $A(P^{(n)}) = \{i\}$, and

$$V_k^{\phi^*}((x,P^{(n-k+1)}),t) = \min_{a\in A(P^{(n-k+1)})} E_a V_{k-1}^{\phi^*}(x,P^{(n-k+1)},t), \tag{2.2}$$

for all $k = 2,3,\cdots,n$, where

$$E_a V_{k-1}^{\phi}(x,P^{(n-k+1)},t) := \sum_{y\in\mathcal{S}} V_{k-1}^{\phi}(y,P^{(n-k+1)} - p_a e_a, t - p_a x) f(y),$$

for $(x,P^{(n-k+1)},t) \in \tilde{\mathcal{S}}_{n-k+1}\times[0,\tau]$, $a \in A(P^{(n-k+1)})$, and $\phi \in \Phi$.

Section 2.4 discusses the exact method to solve the optimality equations given by (2.1)-(2.2); furthermore, the algorithm proposed by [11] to approximate the optimal value function and the optimal policy is studied, and useful properties of this algorithm are presented.

2.4 The Approximate Algorithm

In section 2.3, the n-stage TSSAP is formulated in a form similar to classical dynamic programming problems, and hence, it can be solved by the associated backward recursion algorithm. For notational simplicity, let $((x,P^{(n-k+1)}),t)$ be denoted as $(x,P^{(n-k+1)},t)$ henceforth, for any $((x,P^{(n-k+1)}),t) \in \tilde{\mathcal{S}}_{n-k+1}\times[0,\tau]$. Also, for any $k = 1,2,\cdots,n$, $P^{(n-k+1)} \in \mathcal{W}_{n-k+1}^P$, and $t \in [0,\tau]$, define the function $V_k^{\phi}(.,P^{(n-k+1)},t) : \mathcal{S} \to [0,1]$ to be equal to the target-dependent risk-measure V_k^{ϕ} over the last k time periods under $\phi\,|_k$ where $P^{(n-k+1)}$ and t are fixed. Let $V_k^{\phi}(x,.,t) : \mathcal{W}_{n-k+1}^P \to [0,1]$ and $V_k^{\phi}(x,P^{(n-k+1)},.) : [0,\tau] \to [0,1]$ be defined in a similar fashion. The algorithm proposed by [33] can be modified to compute optimal value functions and optimal policies for the n-stage TSSAP provided that $\mathcal{S}$ is finite. For any $1 \le k \le n$, each given $x \in \mathcal{S}$, and $P^{(n-k+1)} \in \mathcal{W}_{n-k+1}^P$, it follows from this recursive algorithm that $V_k^{\phi^*}(x,P^{(n-k+1)},t)$ is a step distribution

function of t with finite jump points. Let $\{\bar{r}_1, \bar{r}_2, \cdots, \bar{r}_z\}$ be the set of possible rewards that can be obtained during a single arbitrary time period. In addition, let $J_k = \{u_1, u_2, \cdots, u_{j_k}\}$ be the set of all jump points obtained when solving for $V_k^{\phi^*}$ in (2.2). Arrange the $\{u_l + r_i\}$ in ascending order for all $l = 1, 2, \cdots, j_k$ and $i = 1, 2, \cdots, z$ so as to obtain ordered values $v_1 < v_2 < \cdots < v_M$. It is shown by [33] that all the jump points of $V_{k+1}^{\phi^*}$ belong to the set $\{v_1, v_2, \cdots, v_M\}$, and hence, for any given state of the system $(y, P^{(n-k)})$, one only needs to evaluate $V_{k+1}^{\phi^*}(y, P^{(n-k)}, .)$ at the points in $\{v_1, v_2, \cdots, v_M\}$. Furthermore, $V_{k+1}^{\phi^*}(y, P^{(n-k)}, t) = 0$ for $t < v_1$ and $V_{k+1}^{\phi^*}(y, P^{(n-k)}, t) = 1$ for $t \geq v_M$.

The algorithm proposed by [33] quickly becomes computationally inefficient since a growing number of jump points must be considered as one moves backward through each successive stage. In fact, the number of points to consider and the computation time to perform the algorithm grow exponentially as the state space and the action space expand. Therefore, a straightforward computational substitute for solving (2.1)-(2.2) is used in which computations are done on a suitable fixed grid of target values (see [11]). To this end, one can focus on the interval $[0, \tau]$ where τ is the largest target value that is needed to be considered. Note that taking the lower bound of this interval to be zero is well-justified, since task values are assumed to be non-negative. The interval $[0, \tau]$ is then divided into m subintervals using the grid $B = \{t_0, t_1, \cdots, t_m\}$, where $t_0 = 0$, $t_m = \tau$, and $t_i < t_{i+1}$ for $i = 0, 1, \cdots, m-1$. For $k = 1, 2, \cdots, n$ and each $(x, P^{(n-k+1)}) \in \tilde{\mathcal{S}}_{n-k+1}$, the target-dependent risk measure $V_k^{\phi^*}(x, P^{(n-k+1)}, .)$ on $[0, \tau]$ is approximated by a set of values $\left\{(t_0, V_k^{\phi^*}(x, P^{(n-k+1)}, t_0)), (t_1, V_k^{\phi^*}(x, P^{(n-k+1)}, t_1)), \cdots, (t_m, V_k^{\phi^*}(x, P^{(n-k+1)}, t_m))\right\}$. Note that regardless of the iteration index and the time period, the grid set B is kept fixed. This approximate algorithm is referred to as the Grid Method (GM) for solving the n-stage TSSAP.

Let ϕ_m and $V_k^{\phi_m}$ denote the approximate policy and the approximate risk measure over the last k time periods obtained from the GM using the grid set $B = \{t_0, t_1, \cdots, t_m\}$, respectively. Analogous to (2.1)-(2.2), $V_k^{\phi_m}$ is evaluated at the t_i's recursively from $V_{k-1}^{\phi_m}$, and the interpolation of $V_k^{\phi_m}$ between the grid points is performed using any desired approximation. The GM approximation equations are given by

$$V_1^{\phi_m}(x, P^{(n)}, t) := V_1^{\phi^*}(x, P^{(n)}, t), \tag{2.3}$$

for all $(x, P^{(n)}, t) \in \tilde{\mathcal{S}}_n \times [0, \tau]$, and

$$V_k^{\phi_m}(x, P^{(n-k+1)}, t_i) := \min_{a \in A(P^{(n-k+1)})} E_a V_{k-1}^{\phi_m}(x, P^{(n-k+1)}, t_i), \tag{2.4}$$

for all $(x, P^{(n-k+1)}) \in \tilde{\mathcal{S}}_{n-k+1}$, $t_i \in B$, and $k = 2, 3, \cdots, n$, with the interpolation

$$V_k^{\phi_m}(x, P^{(n-k+1)}, t) := V_k^{\phi_m}(x, P^{(n-k+1)}, t_{i-1}), \tag{2.5}$$

if $t \in [t_{i-1}, t_i)$ for some $1 \leq i \leq m$. Observe that by (2.3)-(2.5), $V_k^{\phi_m}(x, P^{(n-k+1)}, .)$ is a step function, with its jump points belonging to the grid set B.

Lemma 2. *The GM defined by* (2.3)-(2.5) *with the grid set* $B = \{t_0, t_1, t_2, \cdots, t_m\}$ *provides a lower bound for the optimal target-dependent risk measure* $V_n^{\phi^*}$. *Therefore,*

$$V_n^{\phi_m}(x, P, t) \leq V_n^{\phi^*}(x, P, t), \tag{2.6}$$

for all $(x, P) \in \tilde{\mathcal{S}}_1$ *and* $t \in [0, \tau]$. *Moreover,* $V_n^{\phi_m}(x, P, .)$ *is a non-decreasing step function on* $[0, \tau]$ *for each* $(x, P) \in \tilde{\mathcal{S}}_1$.

Proof. The proof is by induction on n starting with $n = 2$ tasks as the base case. Observe that for $i = 1, 2, \cdots, m$ and $(x, P^{(n-1)}) \in \tilde{\mathcal{S}}_{n-1}$,

$$V_2^{\phi_m}(x, P^{(n-1)}, t_i) = \min_{a \in A(P)} E_a V_1^{\phi_m}(x, P^{(n-1)}, t_i) = \min_{a \in A(P)} E_a V_1^{\phi^*}(x, P^{(n-1)}, t_i) = V_2^{\phi^*}(x, P^{(n-1)}, t_i), \tag{2.7}$$

where the second equality follows from (2.3). In other words, $V_2^{\phi_m}$ and $V_2^{\phi^*}$ coincide at all the grid points in B, and hence, (2.6) holds true if $t \in B$. Now, assume that $t \in (t_i, t_{i+1})$ for some $0 \leq i \leq m-1$, and note that

$$V_2^{\phi_m}(x, P^{(n-1)}, t) = V_2^{\phi_m}(x, P^{(n-1)}, t_i) = V_2^{\phi^*}(x, P^{(n-1)}, t_i) \leq V_2^{\phi^*}(x, P^{(n-1)}, t),$$

where the first and second equalities follow respectively from (2.5) and (2.7), and the inequality is obtained since $V_2^{\phi^*}(x, P^{(n-1)}, t)$ is a distribution function (and hence, a non-decreasing function) of t on $[0, \tau]$ (see [33]). It is also inferred that $V_2^{\phi_m}(x, P^{(n-1)}, .)$ is a non-decreasing step function due to (2.7) and the fact that $V_2^{\phi^*}(x, P^{(n-1)}, .)$ is non-decreasing on $[0, \tau]$. For the induction step, assume that (2.6) holds true for $n-1$ where $n \geq 3$ and that $V_{n-1}^{\phi_m}$ is a non-decreasing step function on $[0, \tau]$. Now fix an arbitrary $t \in [0, \tau]$, and without loss of generality assume that $t \in [t_i, t_{i+1})$ for some $0 \leq i \leq m-1$. To prove the lemma for n, observe that

$$V_n^{\phi_m}(x, P, t) = V_n^{\phi_m}(x, P, t_i) = \min_{a \in A(P)} E_a V_{n-1}^{\phi_m}(x, P, t_i),$$

and note that since (2.6) holds for $n-1$, it follows that $E_a V_{n-1}^{\phi_m}(x, P, t_i) \leq E_a V_{n-1}^{\phi^*}(x, P, t_i)$ for any $a \in A(P)$. Therefore, $V_n^{\phi_m}(x, P, t_i) \leq V_n^{\phi^*}(x, P, t_i) \leq V_n^{\phi^*}(x, P, t)$, and hence, (2.6) holds true for n. That $V_n^{\phi_m}(x, P, .)$ is a non-decreasing function on $[0, \tau]$, follows from the induction assumption and Proposition 5 in [30]. □

Lemma 3 studies the behavior of the GM defined by (2.3)-(2.5) as the size of the grid set B increases.

Lemma 3. *Consider two sets of breakpoints* $B_1 = \{t_0, t_1, \cdots, t_{m_1}\}$ *and* $B_2 = \{v_0, v_1, \cdots, v_{m_2}\}$ *where* $m_1 < m_2$ *and* $B_1 \subset B_2$ *which implies that* B_2 *provides a finer grid on* $[0, \tau]$. *Let* ϕ_{m_1} *and* ϕ_{m_2} *denote the approximate policies obtained from the GM with grid sets* B_1 *and* B_2, *respectively. Then,*

$$V_n^{\phi_{m_1}}(x, P, t) \leq V_n^{\phi_{m_2}}(x, P, t), \tag{2.8}$$

for all $(x, P, t) \in \tilde{\mathcal{S}}_1 \times [0, \tau]$.

Proof. The proof is by induction on n, with the base of induction starting from $n = 2$. It suffices to show that (2.8) holds for all the elements of B_2 since $V_n^{\phi_{m_1}}(x, P, .)$ and $V_n^{\phi_{m_2}}(x, P, .)$ are step functions whose jump points are elements of B_1 and B_2, respectively. Fix an arbitrary subinterval $[t_i, t_{i+1})$ from B_1 for some $0 \leq i \leq m_1 - 1$, and re-label the breakpoints of B_2 (if any) that lie within this subinterval as $\{\bar{v}_1, \bar{v}_2, \cdots, \bar{v}_{d_i}\}$. Note that

$$V_2^{\phi_{m_2}}(x, P^{(n-1)}, \bar{v}_l) = V_2^{\phi^*}(x, P^{(n-1)}, \bar{v}_l) \geq V_2^{\phi_{m_1}}(x, P^{(n-1)}, \bar{v}_l),$$

for all $l = 1, 2, \cdots, d_i$ where the inequality follows from Lemma 2, and hence, the base of induction is proven. Now, assume that (2.8) holds for $n - 1$ where $n \geq 3$, and observe that

$$V_n^{\phi_{m_2}}(x, P, \bar{v}_l) \geq \min_{a \in A(P)} E_a V_{n-1}^{\phi_{m_1}}(x, P, \bar{v}_l) \geq \min_{a \in A(P)} E_a V_{n-1}^{\phi_{m_1}}(x, P, t_i) = V_n^{\phi_{m_1}}(x, P, t_i) = V_n^{\phi_{m_1}}(x, P, \bar{v}_l),$$

for all $l = 1, 2, \cdots, d_i$ where the first and second inequalities follow respectively from the induction assumption and Lemma 2, and the last equality follows from (2.5). □

Lemma 2 together with Lemma 3 indicate that the GM defined by equations (2.3)-(2.5) with the grid set $B = \{t_0, t_1, \cdots, t_m\}$ provides a lower-bound step-function approximation for the optimal target-dependent risk measure $V_n^{\phi^*}(x, P, .)$ on $[0, \tau]$ for any initial state of the system $(x, P) \in \tilde{\mathcal{S}}_1$; moreover, one obtains better approximations to $V_n^{\phi^*}$ as the size of the grid set B increases. Recall from [33] that $V_n^{\phi^*}(x, P, .)$ is a step distribution function on $[0, \tau]$ for a given $(x, P) \in \tilde{\mathcal{S}}_1$. Now, fix an arbitrary initial state $(x, P) \in \tilde{\mathcal{S}}_1$, and let $J = \{t_1^*, t_2^*, \cdots, t_d^*\}$ and h_j denote the set of all jump points and the j^{th} jump size of $V_n^{\phi^*}(x, P, .)$, respectively. Consider an element of J, t_j^*, that lies within the open subinterval formed by two consecutive elements of B; equivalently, $t_i < t_j^* < t_{i+1}$ for some $1 \leq j \leq d$ and $0 \leq i \leq m-1$. Arbitrarily fix $t \in [t_j^*, t_{i+1})$, and observe that

$$V_n^{\phi^*}(x, P, t) \geq V_n^{\phi^*}(x, P, t_j^*), \tag{2.9}$$

since $t \geq t_j^*$; moreover,

$$V_n^{\phi^*}(x, P, t_j^*) - V_n^{\phi^*}(x, P, t_i) \geq h_j, \tag{2.10}$$

since $t_i < t_j^*$ where t_j^* is the j^{th} jump point of $V_n^{\phi^*}$. Now, note that

$$V_n^{\phi_m}(x, P, t) = V_n^{\phi_m}(x, P, t_i) \leq V_n^{\phi^*}(x, P, t_i), \tag{2.11}$$

which implies that

$$V_n^{\phi^*}(x, P, t) - V_n^{\phi_m}(x, P, t) \geq V_n^{\phi^*}(x, P, t_j^*) - V_n^{\phi_m}(x, P, t) \geq V_n^{\phi^*}(x, P, t_j^*) - V_n^{\phi^*}(x, P, t_i) \geq h_j, \tag{2.12}$$

where the inequalities follow from (2.9), (2.11), and (2.10), respectively . Although it is shown with numerical examples by [11] that the GM (applied to a similar problem in a different context) approximates the optimal

value function extremely well, while reducing the computation time dramatically, (2.12) implies that the gap between $V_n^{\phi^*}(x,P,t)$ and $V_n^{\phi_m}(x,P,t)$ has a lower bound equal to the j^{th} jump size of $V_n^{\phi^*}(x,P,.)$ which is independent of m. Equivalently, no matter how much m increases, there is a difference between the approximation provided by the GM and the optimal risk measure for all $t \in [t_j^*, t_{i+1})$ such that $t_i < t_j^* < t_{i+1}$ for some $1 \leq j \leq d$ and $0 \leq i \leq m-1$. Even by increasing m, such that the smallest grid point that is greater than t_j^* gets closer and closer to t_j^*, this difference cannot be reduced to a value lower than h_j, and hence, the only possible way of eliminating this gap is to pick t_{i+1} such that it coincides with t_j^*. However, choosing the grid set B so that it contains all the jump points of $V_n^{\phi^*}$ is counter-productive since as mentioned before, the exact method to obtain the optimal policy (as defined by [33]) quickly becomes computationally inefficient since more and more jump points must be considered as one moves backward through each successive stage. Therefore, the set B must have a small number of grid points compared to the number of jump points of $V_n^{\phi^*}$ so that the foremost goal of the GM, which is to provide a good approximation in a reasonable amount of time, is not undermined. In other words, having such gaps between the values of $V_n^{\phi^*}$ and $V_n^{\phi_m}$ is inevitable.

It is obvious that this gap results from jump points that exist in the graph of $V_n^{\phi^*}(x,P,.)$ over $[0,\tau]$ (or equivalently, the jump-point discontinuities of $V_n^{\phi^*}(x,P,.)$). One might wonder whether the performance of the GM would be affected positively if these jump points had not existed. Section 2.5 shifts the focus to a n-stage TSSAP with continuous task values, presents sufficient conditions for the existence of a deterministic Markov optimal policy, and provides optimality equations to obtain the optimal policy. Moreover, the behavior of the GM is studied where it is shown that, under certain mild conditions, the optimal target-dependent risk measure has no jump points.

2.5 TSSAP with Continuous Task Values

The task values have been assumed to be discrete random variables. This assumption is relaxed in this section, where a n-stage TSSAP with continuous task values is considered. This results in the state space of the system to be extended to an uncountable set, since the task values can vary in an interval as opposed to a countable set of real numbers (as presumed by the existing literature in this area). Suppose that task values are continuous random variables following a Riemann integrable *pdf* f with support $\mathcal{S} \subseteq [0,+\infty)$. Given a vector of worker values P of size n, the objective is to find a Markov policy $\phi^* \in \Phi_M$ that minimizes the probability of the total reward failing to achieve a target value.

In order to proceed, some notation is introduced. Let $\tilde{\Phi} \subseteq \Phi$ be the set of all policies ϕ such that $V_l^{\phi}(x,P,.)$ is non-decreasing on $[0,\tau]$ for arbitrarily fixed $(x,P) \in \tilde{\mathcal{S}}_1$ and $1 \leq l \leq n$. Under a policy $\phi \in \tilde{\Phi}$ and when the initial state of the system is fixed, the probability of failing to achieve a target value increases as the target value grows. Likewise, let $\tilde{\Phi}_M$ and $\tilde{\Phi}_{DM}$ respectively denote the set of all Markov

and deterministic Markov policies that lie in $\tilde{\Phi}$; equivalently, $\tilde{\Phi}_M := \tilde{\Phi} \cap \Phi_M$ and $\tilde{\Phi}_{DM} := \tilde{\Phi} \cap \Phi_{DM}$. Also, define Δ to be the set of all Markovian decision rules that are Riemann integrable on $\mathcal{S}$, and let the E_a notation, which was introduced in Section 2.4, be modified to suit the continuity assumption in this section, as follows: For a given $(x, P^{(n-k)}, t) \in \tilde{\mathcal{S}}_{n-k} \times [0, \tau]$, $a \in A(P^{(n-k)})$, and an arbitrary decision rule $\delta \in \Delta$, define the operators E_a and E_δ as

$$\begin{aligned} E_a V_k^\phi(x, P^{(n-k)}, t) &:= \int_{\mathcal{S}} V_k^\phi(u, P^{(n-k)} - p_a e_a, t - p_a x) f(u)\, du, \\ E_\delta V_k^\phi(x, P^{(n-k)}, t) &:= \sum_{i \in A(P^{(n-k)})} \delta(i \mid x, P^{(n-k)}, t) E_i V_k^\phi(x, P^{(n-k)}, t), \end{aligned} \tag{2.13}$$

for any $\phi \in \tilde{\Phi}_M$ and $k = 1, 2, \cdots, n-1$. Note that $\delta(i \mid x, P^{(n-k)}, t)$ is the probability that worker i is chosen under decision rule δ given that the current state is $(x, P^{(n-k)}, t)$. Lemma 4 ensures that the operators E_a and E_δ are well-defined.

Lemma 4. *Assume that task values are continuous random variables following a bounded Riemann integrable* pdf f *with interval $\mathcal{S}$ its support. Consider an arbitrary policy $\phi = (\phi_n, \phi_{n-1}, \cdots, \phi_1) \in \tilde{\Phi}_M$ with $\phi_l \in \Delta$ for all $l = 1, 2, \cdots, n$. The following results hold for any $((x, P^{(n-k+1)}), t) \in \tilde{\mathcal{S}}_{n-k+1} \times [0, \tau]$ and $2 \le k \le n$:*

$$\begin{aligned} &1)\ E_{\phi_{n-k+1}} V_{k-1}^\phi(x, P^{(n-k+1)}, t) \text{ is well-defined.} \\ &2)\ V_k^\phi(x, P^{(n-k+1)}, t) = E_{\phi_{n-k+1}} V_{k-1}^\phi(x, P^{(n-k+1)}, t) \end{aligned} \tag{2.14}$$

Proof. Proof of the first statement in (2.14) is by induction on k, and the second equation in (2.14) follows from the first statement. Observe that $V_1^\phi(x, P^{(n)}, t) = P\{p_i x \le t\} = I_{\{p_i x \le t\}}$, for any $\phi \in \tilde{\Phi}_M$ where $A(P^{(n)}) = \{i\}$, and hence, $E_{\phi_{n-1}} V_1^\phi(x, P^{(n-1)}, t)$ is well-defined. The second equation in (2.14) follows directly from conditioning arguments and the Markov property.

For the induction step, assume that $E_{\phi_{n-k+1}} V_{k-1}^\phi(x, P^{(n-k+1)}, t)$ is well-defined. Let $\delta := \phi_{n-k+1}$ and $\pi := \phi|_k$ for notational simplicity, and note that $\pi = (\phi_n, \phi_{n-1}, \cdots, \phi_{n-k+2}, \delta)$. Fix an arbitrary $(x, P^{(n-k+1)}, t) \in \tilde{\mathcal{S}}_{n-k+1} \times [0, \tau]$, let $\bar{P} := P^{(n-k+1)}$, and observe that

$$\begin{aligned} E_\delta V_{k-1}^\phi(x, \bar{P}, t) &= \sum_{a \in A(\bar{P})} \delta(a \mid x, \bar{P}, t) \int_{\mathcal{S}} V_{k-1}^\phi(u, \bar{P} - p_a e_a, t - p_a x) f(u)\, du \\ &= \sum_{a \in A(\bar{P})} \delta(a \mid x, \bar{P}, t) \int_{\mathcal{S}} P_\phi \left\{ R_{k-1} \le t - p_a x \mid \tilde{d}_{n-k+2} = (u, \bar{P} - p_a e_a, t - p_a x) \right\} f(u)\, du \\ &= \sum_{a \in A(\bar{P})} \delta(a \mid x, \bar{P}, t) P_\pi \left\{ R_k \le t \mid \tilde{d}_{n-k+1} = (x, \bar{P}, t), \delta(x, \bar{P}, t) = a \right\} \\ &= V_k^\pi(x, \bar{P}, t), \end{aligned} \tag{2.15}$$

where the third equality follows from the Markov property. This completes the proof of the second statement.

According to (2.13) and (2.15), verifying whether $E_{\phi_{n-k}} V_k^\phi(x, P^{(n-k)}, t)$ is well-defined comes down to

proving that the following integral is well-defined for all $a \in A(P^{(n-k)})$:

$$\int_{\mathcal{S}} E_\delta V_{k-1}^\phi(u, P^{(n-k)} - p_a e_a, t - p_a x) f(u)\, du \tag{2.16}$$

Recall that $E_\delta V_{k-1}^\phi(.) = \sum_i \delta(i \mid .) E_i V_{k-1}^\phi(.)$, and hence, the problem simplifies to showing that

$$\delta(i \mid ., P^{'}, t^{'}) E_i V_{k-1}^\phi(., P^{'}, t^{'}) f(.) \tag{2.17}$$

is Riemann integrable on $\mathcal{S}$ for all i, where $P^{'} := P^{(n-k)} - p_a e_a$ and $t^{'} = t - p_a x$. Fix i and consider the following two cases:

- $\mathcal{S}$ is a bounded interval: Note that $E_i V_{k-1}^\phi(u, P^{'}, t^{'}) = \int_{\mathcal{S}} V_{k-1}^\phi(z, P^{'} - p_i e_i, t^{'} - p_i u) f(z)\, dz$ is well-defined (by the induction assumption) and is a monotone non-increasing function of u, since $\phi \in \tilde{\Phi}_M$ by assumption. Any monotone function on a bounded interval in $\mathbb{R}$ can have at most countably many discontinuity points and is Riemann integrable. Recall that $\delta \in \Delta$, and f is a bounded Riemann integrable function, so (2.17) is Riemann integrable on $\mathcal{S}$.

- $\mathcal{S}$ is an unbounded interval: Observe that $\mathcal{S}$ can be represented as $\mathcal{S} = \cup_{j=1}^{+\infty} I_j$, where I_j's are bounded disjoint intervals. To show that (2.17) is Riemann integrable on $\mathcal{S}$, first show that (2.17) is Riemann integrable on I_j, for $j = 1, 2, \cdots$. To see this, fix j and note that

  $$E_i V_{k-1}^\phi(u, P^{'}, t^{'}) = \int_{\mathcal{S}} V_{k-1}^\phi(z, P^{'} - p_i e_i, t^{'} - p_i u) f(z)\, dz = \sum_{l=1}^{+\infty} \int_{I_l} V_{k-1}^\phi(z, P^{'} - p_i e_i, t^{'} - p_i u) f(z)\, dz,$$

  since $E_i V_{k-1}^\phi$ is well-defined, by the induction assumption. Define

  $$h_l(u) := \int_{I_l} V_{k-1}^\phi(z, P^{'} - p_i e_i, t^{'} - p_i u) f(z)\, dz,$$

  and observe that for any $l \geq 1$, $h_l(u)$ is a monotone function of u and has at most countably many discontinuity points on I_j, which implies that $E_i V_{k-1}^\phi(u, P^{'}, t^{'}) = \sum_{l=1}^{+\infty} h_l(u)$ is Riemann integrable on I_j, and hence, (2.17) is Riemann integrable on I_j, for $j = 1, 2, \cdots$. Now, observe that

  $$0 \leq \int_{I_j} \delta(i \mid u, P^{'}, t^{'}) E_i V_{k-1}^\phi(u, P^{'}, t^{'}) f(u)\, du \leq \int_{I_j} f(u)\, du,$$

  for any $j \geq 1$, which implies that $0 \leq \sum_{j=1}^{+\infty} \int_{I_j} \delta(i \mid u, P^{'}, t^{'}) E_i V_{k-1}^\phi(u, P^{'}, t^{'}) f(u)\, du \leq 1$, and hence, (2.17) is Riemann integrable on $\mathcal{S}$.

□

Before proceeding to Theorem 1, a notation should be introduced. Let $\phi = (\phi_n, \phi_{n-1}, \cdots, \phi_1) \in \tilde{\Phi}_M$

with $\phi_l \in \Delta$ for any $1 \leq l \leq n$ be an arbitrary policy. Based on ϕ, define a deterministic decision rule $\delta^*_{\phi_l} : \tilde{\mathcal{S}}_l \times [0,\tau] \to \mathcal{A}$ such that $\delta^*_{\phi_l}(x, P^{(l)}, t) \in A(P^{(l)})$ and

$$\delta^*_{\phi_l}(x, P^{(l)}, t) := \underset{a \in A(P^{(l)})}{\arg\min}\ E_a V^{\phi}_{n-l}(x, P^{(l)}, t),$$

for all $l = 1, 2, \cdots, n-1$. For $l = n$, let $\delta^*_{\phi_n}$ be the deterministic policy which assigns the last arriving task X_n to the only remaining worker (this is in fact the only admissible policy at time period n). Theorem 1 provides sufficient conditions for the existence of a deterministic Markov optimal policy and specifies the optimality equations to obtain it.

Theorem 1. *Consider the n-stage TSSAP where task values are continuous random variables following a bounded Riemann integrable* pdf *f with an interval $\mathcal{S}$ as its support. The optimality equations*

$$V_1^{\phi^*}(x, P^{(n)}, t) = I_{\{p_i x \leq t\}}, \tag{2.18}$$

where $A(P^{(n)}) = \{i\}$, and

$$V_l^{\phi^*}(x, P^{(n-l+1)}, t) = \min_{a \in A(P^{(n-l+1)})} E_a V_{l-1}^{\phi^*}(x, P^{(n-l+1)}, t), \tag{2.19}$$

for $(x, P^{(n-l+1)}, t) \in \tilde{\mathcal{S}}_{n-l+1} \times [0,\tau]$ and $l = 2, 3, \cdots, n$ yield the optimal policy ϕ^* if

$$\delta^*_{\phi^*_{n-l+1}} \in \Delta, \tag{2.20}$$

for $l = 2, 3, \cdots, n$. Moreover, if (2.20) *holds true, then $\phi^* \in \tilde{\Phi}_{DM}$, and $\delta^*_{\phi^*_k}$ is the optimal decision rule at time period $k = 1, 2, \cdots, n$.*

Proof. The proof is by induction on k. At the final stage (i.e., when the value of the last task is observed), only one worker is remained with value p_i by assumption, and the final task must be matched with this worker independent of the policy applied during the previous stages. Therefore, $V_1^{\phi^*}(x, P^{(n)}, t) = I_{\{p_i x \leq t\}}$, and (2.18) is verified. For the induction step, assume that (2.19) holds for all l where $l \leq k$ with $\pi := \phi^*|_k \in \tilde{\Phi}_M$ and that $\delta^*_{\phi^*_{n-k}} \in \Delta$. Define a policy $\phi_0 := (\pi, \sigma)$ where $\sigma := \delta^*_{\phi^*_{n-k}}$ is a deterministic decision rule. Note that

$$V_{k+1}^{\phi_0}(x, P^{(n-k)}, t) = E_\sigma V_k^{\pi}(x, P^{(n-k)}, t) = E_\sigma V_k^{\phi^*}(x, P^{(n-k)}, t) = \min_{a \in A(P^{(n-k)})} E_a V_k^{\phi^*}(x, P^{(n-k)}, t), \tag{2.21}$$

where the first, the second, and the last equalities follow from Lemma 4, the definition of π, and the definition of σ, respectively. Moreover, $\phi^*|_k \in \tilde{\Phi}_M$ by the induction assumption, which implies that $E_a V_k^{\phi^*}(x, P^{(n-k)}, .)$ is non-decreasing for any $a \in A(P^{(n-k)})$. Therefore, it follows that $V_{k+1}^{\phi_0}$ is a non-decreasing function of t

which implies that $\phi_0 \in \tilde{\Phi}_M$. Observe that

$$V_{k+1}^{\phi^*}(x, P^{(n-k)}, t) = \inf_{\phi \in \Phi_M} V_{k+1}^{\phi}(x, P^{(n-k)}, t) \leq V_{k+1}^{\phi_0}(x, P^{(n-k)}, t) = \min_{a \in A(P^{(n-k)})} E_a V_k^{\phi^*}(x, P^{(n-k)}, t). \tag{2.22}$$

Now, consider an arbitrary policy $\phi = (\phi_n, \phi_{n-1}, \cdots, \phi_{n-k}) \in \Phi_M$, let $\bar{P} := P^{(n-k)}$, and obtain the following:

$$\begin{aligned}
V_{k+1}^{\phi}(x, \bar{P}, t) &= \sum_{i \in A(\bar{P})} \phi_{n-k}(i \mid x, \bar{P}, t) P_\phi \left\{ R_{k+1} \leq t \middle| \tilde{d}_{n-k} = (x, \bar{P}, t), \phi_{n-k}(x, \bar{P}, t) = i \right\} \\
&= \sum_{i \in A(\bar{P})} \phi_{n-k}(i \mid x, \bar{P}, t) P_\phi \left\{ R_k \leq t - p_i x \middle| \tilde{d}_{n-k} = (x, \bar{P}, t), \phi_{n-k}(x, \bar{P}, t) = i \right\} \\
&= \sum_{i \in A(\bar{P})} \phi_{n-k}(i \mid x, \bar{P}, t) P_\phi \left\{ R_k \leq t - p_i x \middle| \tilde{d}_{n-k+1} = (X_{n-k+1}, \bar{P} - p_i e_i, t - p_i x) \right\} \\
&\geq \sum_{i \in A(\bar{P})} \phi_{n-k}(i \mid x, \bar{P}, t) P_{\phi^*} \left\{ R_k \leq t - p_i x \middle| \tilde{d}_{n-k+1} = (X_{n-k+1}, \bar{P} - p_i e_i, t - p_i x) \right\} \\
&= \sum_{i \in A(\bar{P})} \phi_{n-k}(i \mid x, \bar{P}, t) E_i V_k^{\phi^*}(x, \bar{P}, t) \\
&\geq \min_{a \in A(\bar{P})} E_a V_k^{\phi^*}(x, \bar{P}, t),
\end{aligned}$$

which implies that $V_{k+1}^{\phi^*}(x, P^{(n-k)}, t) \geq \min_{a \in A(P^{(n-k)})} E_a V_k^{\phi^*}(x, P^{(n-k)}, t)$. Combining this with (2.22) yields

$$V_{k+1}^{\phi^*}(x, P^{(n-k)}, t) = V_{k+1}^{\phi_0}(x, P^{(n-k)}, t) = \min_{a \in A(P^{(n-k)})} E_a V_k^{\phi^*}(x, P^{(n-k)}, t),$$

where $\phi^*|_{k+1} := \phi_0 = (\phi^*|_k, \delta^*_{\phi^*_{n-k}}) \in \tilde{\Phi}_{DM}$. □

Theorem 2 provides a sufficient condition for (2.20) to hold and presents useful properties of the optimal policy and the optimal value function under this condition.

Theorem 2. *Consider the n-stage TSSAP where task values are continuous random variables following a bounded Riemann integrable pdf f that has interval $\mathcal{S}$ as its support. Then:*

- (2.20) *is satisfied, and hence, the optimal policy $\phi^* \in \Phi_{DM}$ is obtained by* (2.18)-(2.19).
- *The optimal decision rule at time period k is right-continuous in the value of the current task and in the current target level, for $k = 1, 2, \cdots, n-1$.*
- *$V_n^{\phi^*}(x, P, t)$ is continuous in x and in t and is a non-decreasing function of t. Moreover, it is a distribution function in t if $\mathcal{S}$ is bounded.*

Proof. The proof is by induction on n starting with $n = 2$ tasks as the base case. Fix an arbitrary task value $x \in \mathcal{S}$ and a vector of worker values $P^{(n-1)}$ at time period $n = 2$, and assume without loss of generality that

$A(P^{(n-1)}) = \{1, 2\}$. Note that if $E_i V_1^{\phi^*}$ is a continuous function of t for $i = 1, 2$ and $\delta^*_{\phi^*_{n-1}} \in \Delta$, then $V_2^{\phi^*}$ is continuous in t due to (2.19) and the fact that it is the minimum over a finite set of continuous functions. Observe that

$$E_2 V_1^{\phi^*}(x, P^{(n-1)}, t) = \int_{\mathcal{S} \cap [0, \frac{t - p_2 x}{p_1}]} f(u)\, du = P\left\{ X_2 \in \mathcal{S} \cap [0, \frac{t - p_2 x}{p_1}] \right\},$$

which implies that $E_2 V_1^{\phi^*}$ is continuous in t. Similarly, $E_1 V_1^{\phi^*}$ is shown to be continuous in t; therefore, the optimal decision rule at this time period (i.e., when X_{n-1} arrives) is right-continuous in t when $(x, P^{(n-1)})$ is kept fixed. Following the same argument as above results in $E_1 V_1^{\phi^*}$ and $E_2 V_1^{\phi^*}$ being continuous functions of x on $\mathcal{S}$. This implies that (2.20) is satisfied (i.e., $\delta^*_{\phi^*_{n-1}} \in \Delta$); thus, the optimality equation (2.19) is used to derive $\delta^*_{\phi^*_{n-1}}$ and $V_2^{\phi^*}$. It also follows that $V_2^{\phi^*}$ is continuous in x and in t and is a distribution function in t. For the induction step, assume that Theorem 2 holds true for a TSSAP with $n-1$ tasks where $n \geq 3$. To prove the theorem for n, fix $(x, P) \in \tilde{\mathcal{S}}_1$ and $i \in A(P)$, and assume that the sequence $\{t_j\}$ converges to t_0 as $j \to +\infty$, where $t_0 \in [0, \tau]$ is arbitrarily fixed. To prove that $E_i V_{n-1}^{\phi^*}$ is continuous in t, consider the following two cases:

- $\mathcal{S}$ is a bounded interval: Observe that for any $u \in \mathcal{S}$,

$$\lim_{j \to +\infty} V_{n-1}^{\phi^*}(u, P - p_i e_i, t_j - p_i x) = V_{n-1}^{\phi^*}(u, P - p_i e_i, t_0 - p_i x), \tag{2.23}$$

  by the induction assumption that $V_{n-1}^{\phi^*}$ is continuous in its third argument. Let M be an upper bound of f on $\mathcal{S}$ and note that

$$0 \leq V_{n-1}^{\phi^*}(u, P - p_i e_i, t_j - p_i x) f(u) \leq M, \tag{2.24}$$

  for all $u \in \mathcal{S}$ and $j \geq 1$, since $V_{n-1}^{\phi^*}$ is a distribution function by the induction assumption. Hence, the bounded convergence theorem implies that $\lim_{j \to +\infty} E_i V_{n-1}^{\phi^*}(x, P, t_j) = E_i V_{n-1}^{\phi^*}(x, P, t_0)$ by (2.23), (2.24), and the fact that the right-hand side of (2.23) is integrable over $\mathcal{S}$ by the induction assumption. Therefore, $E_i V_{n-1}^{\phi^*}(x, P, t)$ is continuous in t.

- $\mathcal{S}$ is an unbounded interval: Observe that $\mathcal{S}$ can be represented as $\mathcal{S} = \cup_{j=1}^{+\infty} I_j$, where I_j's are bounded disjoint intervals. Note that $h_j(t) := \int_{I_j} V_{n-1}^{\phi^*}(u, P - p_i e_i, t - p_i x) f(u)\, du$ is continuous in t for all j, by the bounded convergence theorem. Also, $0 \leq h_j(t) \leq \int_{I_j} f(u)\, du$, for any $t \in [0, \tau]$ and $j = 1, 2, \cdots$, where $\sum_{j=1}^{+\infty} \int_{I_j} f(u)\, du = 1 < +\infty$. The Weierstrass M-test implies that $\sum_{j=1}^{+\infty} h_j(t)$ converges uniformly on $[0, \tau]$, and hence, $E_i V_{n-1}^{\phi^*}(x, P, t)$ is a continuous function of t.

Likewise, it is proven for each $i \in A(P)$ that $E_i V_{n-1}^{\phi^*}(x, P, t)$ is continuous in x. Continuity of $E_i V_{n-1}^{\phi^*}$ in x and in t for all $i \in A(P)$ implies that the optimal decision rule upon the arrival of the first task (i.e., $\delta^*_{\phi_1}$) is right-continuous in both x and t. Therefore, the optimality equation (2.19) is used to derive $V_n^{\phi^*}$, and

hence, $V_n^{\phi^*}$ is continuous in both t and x since the action space $A(P)$ is finite. It also follows that $V_n^{\phi^*}$ is a distribution function in t by the induction assumption (see Proposition 5 in [30]) if $\mathcal{S}$ is bounded. □

Remark. The results of Lemma 4, Theorem 1, and Theorem 2 can be generalized to the case where the support of f is an open subset of $[0, +\infty)$ or a countable union of disjoint intervals in $[0, +\infty)$ (Recall that every open set in $\mathbb{R}$ can be represented as a countable union of disjoint bounded open intervals). The proof is similar to that presented for Lemma 4.

Lipschitz continuity of the optimal value function is proven in Theorem 3 for continuous task values with bounded integrable probability distribution functions.

Theorem 3. *Consider the n-stage TSSAP where task values are continuous random variables following a bounded Riemann integrable pdf f that has an interval $\mathcal{S}$ as its support. For any arbitrarily fixed pair $(x, P) \in \tilde{\mathcal{S}}_1$, $V_n^{\phi^*}(x, P, t)$ is Lipschitz continous in t:*

$$\left|V_n^{\phi^*}(x, P, t) - V_n^{\phi^*}(x, P, s)\right| \leq C \mid t - s \mid, \tag{2.25}$$

for $s, t \in [0, \tau]$ where C is a positive constant, independent of x.

Proof. The proof is by induction on n starting with $n = 2$ tasks as the base case. Arbitrarily fix $(x, P^{(n-1)}) \in \tilde{\mathcal{S}}_{n-1}$, and assume without loss of generality that $A(P^{(n-1)}) = \{1, 2\}$. Let $s, t \in [0, \tau]$ with $s \leq t$ and $\tilde{P} := P^{(n-1)} - p_2 e_2$, and observe that

$$\begin{aligned}
\left|E_2 V_1^{\phi^*}(x, P^{(n-1)}, t) - E_2 V_1^{\phi^*}(x, P^{(n-1)}, s)\right| &\leq \int_{\mathcal{S}} \left|V_1^{\phi^*}(u, \tilde{P}, t - p_2 x) - V_1^{\phi^*}(u, \tilde{P}, s - p_2 x)\right| f(u)\, du \\
&= \int_{\mathcal{S} \cap (\frac{s - p_2 x}{p_1}, \frac{t - p_2 x}{p_1}]} f(u)\, du \\
&\leq \frac{M}{p_{min}}(t - s),
\end{aligned}$$

where M is an upper bound of f on $\mathcal{S}$ and $p_{min} := \min_{i \in \{1, 2, \cdots, n\}} p_i$. A similar argument can be made for $E_1 V_1^{\phi^*}$, and hence, it follows that $|V_2^{\phi^*}(x, P^{(n-1)}, t) - V_2^{\phi^*}(x, P^{(n-1)}, s)| \leq \frac{M}{p_{min}}|t - s|$, for all $(x, P^{(n-1)}) \in \tilde{\mathcal{S}}_{n-1}$ and $s, t \in [0, \tau]$. For the induction step, assume that $V_{n-1}^{\phi^*}(x, P^{(2)}, t)$ is Lipschitz continous on $[0, \tau]$ with the Lipschitz constant $\frac{M}{p_{min}}$ for all $(x, P^{(2)}) \in \tilde{\mathcal{S}}_2$ and some $n \geq 3$. To show that (2.25) holds for n, fix $(x, P) \in \tilde{\mathcal{S}}_1$, and observe that

$$\begin{aligned}
\left|E_i V_{n-1}^{\phi^*}(x, P, t) - E_i V_{n-1}^{\phi^*}(x, P, s)\right| &\leq \int_{\mathcal{S}} \left|V_{n-1}^{\phi^*}(u, P - p_i e_i, t - p_i x) - V_{n-1}^{\phi^*}(u, P - p_i e_i, s - p_i x)\right| f(u)\, du \\
&\leq \frac{M}{p_{min}}|t - s|,
\end{aligned}$$

for all $i \in A(P)$ and $s, t \in [0, \tau]$, which implies that $|V_n^{\phi^*}(x, P, t) - V_n^{\phi^*}(x, P, s)| \leq \frac{M}{p_{min}}|t - s|$, due to the finiteness of the action space. □

Corollary 1 presents an interesting property of the optimal value function.

Corollary 1. *Consider the n-stage TSSAP where task values follow a bounded Riemann integrable pdf f with an interval $\mathcal{S}$ as its support. For any sequence $\{t_j\}_{j=1}^{+\infty}$ converging to an arbitrarily fixed $t \in [0,\tau]$, $V_n^{\phi^*}(x,P,t_j)$ converges uniformly to $V_n^{\phi^*}(x,P,t)$ on $\mathcal{S}$ as $j \to +\infty$; i.e.,*

$$\sup_{x\in\mathcal{S}} \left|V_n^{\phi^*}(x,P,t_j) - V_n^{\phi^*}(x,P,t)\right| \to 0 \quad as \quad j \to +\infty.$$

Proof. The proof follows from Theorem 3 and the Cauchy criterion for uniform convergence and is eliminated due to simplicity. □

Before proceeding to Lemma 5, the GM presented in Section 2.4 for the discrete case is generalized to the continuous case using (2.3)-(2.5), where $E_i V_l^{\phi_m}$ is defined in (2.13). Lemma 5 proves that the operator E_i is well-defined for the GM.

Lemma 5. *Consider the n-stage TSSAP with continuous task values following a bounded Riemann integrable pdf f with an interval $\mathcal{S}$ as its support. Let $B = \{t_0, t_1, \cdots, t_m\}$ with $t_0 = 0$ and $t_m = \tau$ be a grid set for the GM. For $l = 1, 2, \cdots, n$, $V_l^{\phi_m}(., P^{(n-l+1)}, t)$ and $V_l^{\phi_m}(x, P^{(n-l+1)}, .)$ are non-increasing and non-decreasing functions, respectively. Moreover, $E_i V_l^{\phi_m}$ is well-defined for $l = 1, 2, \cdots, n-1$.*

Proof. The proof of $V_l^{\phi_m}$ being a monotone function on $\mathcal{S}$ and on $[0,\tau]$ is by induction on l. That the operator $E_i V_l^{\phi_m}$ is well-defined follows from $V_l^{\phi_m}$ being a monotone function on $\mathcal{S}$. The proof is eliminated due to simplicity. □

Note that the results of Lemma 2 and Lemma 3 are easily generalized to the GM defined for the continuous case. For a n-stage TSSAP with continuous task values, Proposition 1 studies the behavior of the GM as the grid on $[0,\tau]$ becomes finer.

Proposition 1. *Consider the n-stage TSSAP with a given vector of workers P and continuous task values following a bounded Riemann integrable pdf f that has an interval $\mathcal{S}$ as its support. For a grid set $B = \{t_0^m, t_1^m, \cdots, t_m^m\}$ where $t_i^m := \frac{\tau}{m} i$, $V_n^{\phi_m}(x,P,t)$ converges to $V_n^{\phi^*}(x,P,t)$ uniformly on $\mathcal{S} \times [0,\tau]$ with order one as $m \to +\infty$; i.e.,*

$$\sup_{x\in\mathcal{S}, t\in[0,\tau]} \left|V_n^{\phi^*}(x,P,t) - V_n^{\phi_m}(x,P,t)\right| = O\left(m^{-1}\right). \tag{2.26}$$

Proof. Observe that by (2.3) and as in the discrete case, $V_2^{\phi^*}$ and $V_2^{\phi_m}$ coincide at the breakpoints; therefore, $V_2^{\phi^*}(x, P^{(n-1)}, t_i^m) = V_2^{\phi_m}(x, P^{(n-1)}, t_i^m)$, for $i = 0, 1, 2, \cdots, m$ and $(x, P^{(n-1)}) \in \tilde{\mathcal{S}}_{n-1}$. Fix an arbitrary $(x, P^{(n-1)}) \in \tilde{\mathcal{S}}_{n-1}$ and $t \in [0,\tau]$, and without loss of generality assume that t belongs to the $(k+1)^{th}$ interval defined by B on $[0,\tau]$ for some $0 \le k \le m-1$ (i.e., $t \in [t_k^m, t_{k+1}^m)$). Note that the GM provides a

lower bound approximation for the optimal value function; therefore,

$$0 \leq V_2^{\phi^*}(x, P^{(n-1)}, t) - V_2^{\phi_m}(x, P^{(n-1)}, t) \leq V_2^{\phi^*}(x, P^{(n-1)}, t_{k+1}^m) - V_2^{\phi^*}(x, P^{(n-1)}, t_k^m) \leq C\frac{\tau}{m},$$

where the second inequality follows from Lemma 5 (specifically, the fact that $V_2^{\phi^*}(x, P^{(n-1)}, .)$ is a non-decreasing function), and C is the Lipschitz constant defined in Theorem 3; hence, (2.26) is satisfied for a problem of size $n = 2$. For the induction step, assume that for some $n \geq 3$:

$$0 \leq V_{n-1}^{\phi^*}(u, P^{(2)}, t_{j+1}^m) - V_{n-1}^{\phi_m}(u, P^{(2)}, t_j^m) \leq (n-2)C\frac{\tau}{m},$$

for all $u \in \mathcal{S}$, $P^{(2)} \in \mathcal{W}_2^P$, and $j = 0, 1, \cdots, m-1$. To prove (2.26) for n, observe that

$$\begin{aligned} 0 &\leq V_n^{\phi^*}(x, P, t) - V_n^{\phi_m}(x, P, t) \\ &\leq V_n^{\phi^*}(x, P, t_{k+1}^m) - V_n^{\phi_m}(x, P, t_k^m) \\ &= \min_{i \in A(P)} E_i V_{n-1}^{\phi^*}(x, P, t_{k+1}^m) - \min_{i \in A(P)} E_i V_{n-1}^{\phi_m}(x, P, t_k^m). \end{aligned} \tag{2.27}$$

Since $t_{k+1}^m - t_k^m = \frac{\tau}{m}$ (or equivalently, $(t_{k+1}^m - p_i x) - (t_k^m - p_i x) = \frac{\tau}{m}$), it follows that there exists $k^{'} < k$ such that $t_k^m - p_i x \in [t_{k^{'}}^m, t_{k^{'}+1}^m]$ and $t_{k+1}^m - p_i x \in [t_{k^{'}+1}^m, t_{k^{'}+2}^m)$. Therefore,

$$E_i V_{n-1}^{\phi^*}(x, P, t_{k+1}^m) = \int_{\mathcal{S}} V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k+1}^m - p_i x) f(u)\, du \leq \int_{\mathcal{S}} V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+2}^m) f(u)\, du,$$

and

$$E_i V_{n-1}^{\phi_m}(x, P, t_k^m) = \int_{\mathcal{S}} V_{n-1}^{\phi_m}(u, P - p_i e_i, t_k^m - p_i x) f(u)\, du = \int_{\mathcal{S}} V_{n-1}^{\phi_m}(u, P - p_i e_i, t_{k^{'}}^m) f(u)\, du,$$

which leads to

$$\begin{aligned} E_i V_{n-1}^{\phi^*}(x, P, t_{k+1}^m) - E_i V_{n-1}^{\phi_m}(x, P, t_k^m) &\leq \int_{\mathcal{S}} (V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+2}^m) - V_{n-1}^{\phi_m}(u, P - p_i e_i, t_{k^{'}}^m)) f(u)\, du \\ &= \int_{\mathcal{S}} (V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+2}^m) - V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+1}^m)) f(u)\, du \\ &+ \int_{\mathcal{S}} (V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+1}^m) - V_{n-1}^{\phi_m}(u, P - p_i e_i, t_{k^{'}}^m)) f(u)\, du. \end{aligned} \tag{2.28}$$

Now, recall from Theorem 3 that $V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+2}^m) - V_{n-1}^{\phi^*}(u, P - p_i e_i, t_{k^{'}+1}^m) \leq C\frac{\tau}{m}$, for all $u \in \mathcal{S}$. Moreover, the second term on the right-hand side of equation (2.28) can be bounded above by the induction assumption. Therefore,

$$0 \leq E_i V_{n-1}^{\phi^*}(x, P, t_{k+1}^m) - E_i V_{n-1}^{\phi_m}(x, P, t_k^m) \leq (n-1)C\frac{\tau}{m}, \tag{2.29}$$

for all $i \in A(P)$, which implies that

$$0 \leq V_n^{\phi^*}(x, P, t) - V_n^{\phi_m}(x, P, t) \leq (n-1)C\frac{\tau}{m},$$

by (2.27) and from the finiteness of the action space. □

Proposition 1 establishes the uniform convergence of the approximate value function obtained by the GM to the optimal value function as the grid on $[0, \tau]$ becomes finer. However, recall that for the n-stage TSSAP with discrete task values, there are intervals within $[0, \tau]$ in which a lower bound exists on the difference between the approximate and the optimal value function. As shown in Section 2.4, this lower bound is a constant and unaffected by increases in m (i.e., the number of grid points); equivalently, for any value of m which results in a reasonable computation time for the GM, there always exist intervals within $[0, \tau]$ with gaps (greater than a given constant) between the approximate and optimal value functions. Therefore, the GM has a better performance when applied to the n-stage TSSAP with countinuous task values. Section 2.6 provides numerical results to compare the performance of the SSAP and TSSAP.

2.6 Numerical Results

This section compares the performance of the optimal policies obtained from the SSAP and TSSAP, using a numerical example. Consider a stochastic sequential assignment problem with $n = 10$ tasks arriving sequentially at each time period to be allocated to the available resources. Assume that the task values follow a Binomial distribution with parameters $(4, 0.3)$ and the vector of worker values is given by $P = (10, 50, 100, 150, 250, 400, 540, 600, 750, 950)$. We solve the TSSAP for each target value within the interval $[3000, 13000]$ with a step size of 50. For each fixed target value, a total of $s = 1000$ TSSAP's are solved by simulating the arriving task values with the given Binomal distribution, and for every one of the 1000 problems simulated, a SSAP is solved as well. For a fixed target value τ, let r_T^τ and r_S^τ denote the number of times (out of a 1000 simulations) that the TSSAP and the SSAP yield a total reward lower than τ. Figure 6.1 depicts the ratio $\frac{r_S^\tau}{r_T^\tau}$ as a function of τ. As it can be seen from the figure, the optimal policy from the TSSAP performs significantly better than that of the SSAP for target values that are below or around the $E_f[X]\sum_{i=1}^n p_i$ (where $E_f[X]$ is the expected value of X_j and p_i is the success rate of the i^{th} worker). As the target value increases, the ratio $\frac{r_S^\tau}{r_T^\tau}$ decreases but stabilizes at one (which is intuitive).

2.7 Conclusion

Chapter 2 studies the SSAP under the threshold criterion, which attempts to minimize the probability of the total reward (obtained from the sequential assignment of tasks to available workers) failing to achieve

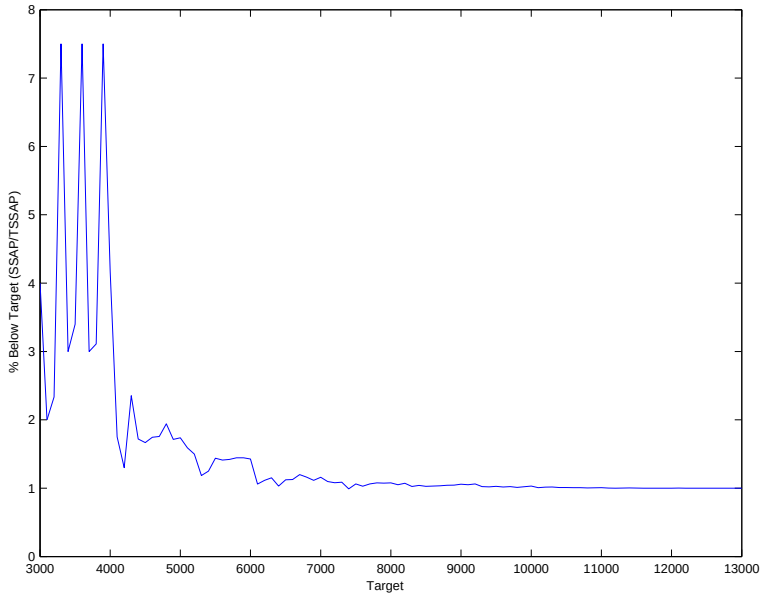

Figure 2.1: Comparing the optimal policy of SSAP vs. TSSAP

a specified target value. The problem is modelled as a MDP for discrete task values and is then extended to the case where the state space of arriving tasks is uncountable (i.e., task values are considered to be continuous random variables). Sufficient conditions for the existence of a deterministic Markov optimal policy are derived along with fundamental properties of the optimal value function. An algorithm (referred to here as GM) is introduced to approximate the optimal value function and the optimal policy, since the problem becomes computationally inefficient and intractable as the number of arriving tasks increases. The behavior of GM is analyzed for the countable and the uncountable state space cases, and convergence of the approximate value function (obtained by GM) to the optimal value function is established.

It is assumed here that the underlying distribution function of task values is given beforehand, and further research is required to address the TSSAP in which task values follow a probability distribution with unknown parameters. In addition, a possible extension of the TSSAP is to the case where the total number of tasks is unknown until after the final arrival and follows a generic probability distribution. Other challenges include shifting one's attention from the IID sequence of tasks to a more general case with dependent task values and/or considering an infinite sequence of arriving tasks. Moreover, another research direction is extending the main results in this chapter, which are obtained for the SSAP, to a more general MDP framework, where the action space is not necessarily finite.

Chapter 3

Limiting Behavior of the Stochastic Sequential Assignment Problem

3.1 Introduction

Consider the sequential stochastic assignment problem (SSAP) introduced in [12], where n workers are available to perform n tasks. The IID tasks arrive in sequence with the random variable X_j denoting the j^{th} task value. A value (success rate) p_i is associated with each worker, and whenever the i^{th} worker is assigned to the j^{th} task, that worker becomes unavailable for future assignments, with $p_i x_j$ denoting the expected reward due to this assignment. The objective is to assign these n workers to n tasks so as to maximize the expected total reward. It is shown in [12] that there exists numbers

$$-\infty = a_{0,n} \leq a_{1,n} \leq a_{2,n} \leq \cdots \leq a_{n,n} = +\infty,$$

such that the optimal choice in the initial stage is to assign the ith best available worker if the random variable X_1 falls within the i^{th} highest interval. The assumption that the number of tasks equals the number of workers can be relaxed as follows. Let m denote the number of tasks, while n is the number of workers. If $m > n$, then we add $m - n$ phantom workers with success rates of 0, while if $m < n$, the $n - m$ workers with the smallest values are dropped so that only those m workers with the highest success rates can be chosen. The SSAP has applications in several areas. For example, [23] studies a variation of the SSAP in aviation security screening systems. Passengers, arriving sequentially, are assigned a perceived risk level and must be screened by the appropriate security device (among a set of available devices) to maximize the expected total security. In addition, [31] addresses the problem of allocating sequentially-arriving kidneys to patients on a waiting list. Another application of the SSAP is the asset selling problem, studied in [5], where one needs to choose the best offers out of a sequence of bids from potential buyers.

Implementing the optimal assignment policy for the SSAP, as described in [12], involves calculating a new set of breakpoints upon the arrival of each task. The computation of these breakpoints is cumbersome for large scale problems such as aviation screening, where passengers are assigned to aviation security resources based on their perceived threat levels [23]. Assigning passengers to security devices by re-calculating the breakpoints, every time a passenger arrives, is not practical. Therefore, this work is concerned with the

limiting behavior of the $\{a_{i,n}\}$ as n approaches infinity, with the hope of obtaining simpler solutions which are implementable in the real-world problems. Consider the SSAP with n tasks and k (fixed) worker categories, where the i^{th} worker-category consists of r_i workers each with value p_i such that $\sum_{i=1}^{k} r_i = n$. The goal is to maximize the expected reward per task as $n \to +\infty$. Two versions of this problem are studied here. First, it is assumed that the tasks are IID with a known distribution function. The second problem then considers the case that the task values are random variables coming from r different distributions, where the successive distributions are governed by an ergodic Markov chain. Once a task arrives in a given time period, its value is observed; however, the distribution it comes from is unobservable. Simple stationary policies are presented for both problems, which achieve the optimal long-run expected reward per task. These policies consist of $k-1$ fixed (time-independent) breakpoints, as opposed to the policy described by [12] in which the number of breakpoints increases with n and the (time-dependent) breakpoints are recalculated each time a task arrives. Furthermore, convergence rate of the expected reward per task to the optimal value under this stationary policy is obtained for both problems.

The limiting behavior of SSAP has been addressed in [7] and [4]. In particular, [7] characterizes a threshold optimal policy for a secretary problem with IID tasks (a special case of SSAP where the vector of workers' success rates consists of only zeros and ones) that achieves the optimal expected reward per task as $n \to +\infty$. In [4], a secretary problem is studied where the values of successive tasks come from r different known distributions; however, [4] assumes that once a task arrives, both its value and its distribution are observed. The assumption that the underlying distribution of the arriving tasks is known apriori (and observable) does not hold in most real-world problems. The present work deviates from the existing literature since it considers the task distributions to be unobservable, forming a hidden Markov chain. Moreover, the workers' success rates are allowed to take on any arbitrary value, as opposed to only zeros and ones (as in the secretary problem studied in the literature). In fact, this model incorporates both dependency and uncertainty into the stochastic sequential assignment problem, where the former is due to the Markov chain and the latter is due to not observing the distribution function of task values. For this model, the invariant distribution of the ergodic Markov chain (governing the distribution functions) is used to derive a simple time-independent policy which achieves optimality in the long-run. However, the SSAP literature has mainly taken a different approach when dealing with the uncertainty due to the distribution function of task values in the sense that one or more parameters of the distribution function are assumed to be unknown, with a given prior distribution. For example, [5] addresses the problem of uncertainty in the generalized house selling problem, where a Bayesian approach to updating the unknown parameters in the distribution of house bids is applied, resulting in time-dependent optimal policies. Other examples include [1] and [10]. It is noteworthy to mention that although [7] has laid out the framework for the analysis of the limiting

behavior of the SSAP, the proof approach applied in this work is different from [7] and [4], mainly since this works generalizes their work to a problem with more than two worker categories. This will be discussed more specifically in Section 3.2 and Section 3.3.

The chapter is organized as follows: Section 3.2 studies the first model in which task values are IID with a known observable distribution, where an optimal assignment policy and convergence-rate results are presented. The problem of unobservable task distributions, forming a hidden Markov chain, is addressed in Section 3.3.

3.2 The Model: Observable Task Distributions

Consider the SSAP with n tasks and k (fixed) worker categories, where the i^{th} worker-category consists of r_i workers each with value p_i. Let [.] denote the floor function where $[y] := \max\left\{m \in \mathbb{Z} \middle| m \leq y\right\}$. Moreover, let π_i be the fraction of total number of workers that belong to categories $i+1$ to k, for $i = 0, 1, 2, \cdots, k-1$. For simplicity, the i^{th} worker category is referred to as the type-i workers, and the size of the i^{th} category is given by

$$r_i = [n\pi_{i-1}] - [n\pi_i] \quad \text{for } i = 1, 2, \cdots, k,$$

where $\pi_0 = 1$, $\pi_k = 0$, and $\pi_{i+1} < \pi_i$ for $i = 0, 1, \cdots, k-1$. Also, assume that $p_{i+1} < p_i$ for $i = 1, 2, \cdots, k$. The optimal expected total reward for this SSAP is given by

$$\sum_{j=1}^{k} \left(p_j \sum_{i=[n\pi_j]+1}^{[n\pi_{j-1}]} a_{i,n+1} \right),$$

according to [12], where the sequence of arriving tasks is IID. Moreover, it is proven in [7] that

$$\lim_{n\to+\infty} \frac{1}{n} \sum_{i=1}^{[n\pi]} a_{i,n+1} = \int_{-\infty}^{F^{-1}(\pi)} xF(dx),$$
$$\lim_{n\to+\infty} \frac{1}{n} \sum_{i=[n\pi]+1}^{n} a_{i,n+1} = \int_{F^{-1}(\pi)}^{+\infty} xF(dx),$$

for any $0 < \pi < 1$ if F (the distribution function of X) is continuous. Therefore, the optimal long-run expected reward per task for this SSAP with k fixed worker categories is given by

$$\lim_{n\to+\infty} \frac{1}{n} \sum_{j=1}^{k} \left(p_j \sum_{i=[n\pi_j]+1}^{[n\pi_{j-1}]} a_{i,n+1} \right) = \sum_{j=1}^{k} \left(p_j \int_{F^{-1}(\pi_j)}^{F^{-1}(\pi_{j-1})} xF(dx) \right), \tag{3.1}$$

with $F^{-1}(1) = +\infty$ and $F^{-1}(0) = -\infty$. For simplicity, let r^* denote the right-hand side of equation (3.1). A stationary policy with one breakpoint is proposed in [7], and to prove its optimality, an infeasible policy is considered, the long-run expected reward of which can be computed. It is shown in [7] that only two possible scenarios can happen and that the difference between the long-run expected reward per task for this

infeasible policy and the proposed policy converges to zero as $n \to +\infty$, for both possible cases. However, this can be done since only two worker categories exist, and when dealing with the general k-worker-class problem, the number of possible scenarios to consider gets intractable. Therefore, the proof approach is altered to some extent in this dissertation.

Consider a policy that assigns the j^{th} task to a type-i worker if $X_j \in (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$. In what follows, it is proven that this policy achieves the optimal long-run expected reward per job given in (3.1). Note that this policy consists of $k-1$ (time-independent) fixed breakpoints, where k does not change as $n \to +\infty$, as opposed to the policy described by [12] in which the number of breakpoints increases with n and the breakpoints are recalculated at each time period.

Let $\mathcal{A}_i := (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$ for $i = 1, 2, \cdots, k$. A task X_j is labeled type-i if $X_j \in \mathcal{A}_i$. For a fixed n, let $U_{r_i}^{(n)}$ denote the number of tasks that should arrive until r_i tasks of type i are obtained and define $U^{(n)} := \min\left\{U_{r_1}^{(n)}, U_{r_2}^{(n)}, \cdots, U_{r_k}^{(n)}\right\}$. Observe that $U_{r_i}^{(n)}$ follows a negative binomial distribution with parameters $(r_i, \pi_{i-1} - \pi_i)$, since tasks are assumed to be IID. Proposition 2 presents a useful property of $U^{(n)}$.

Proposition 2.

$$\lim_{n \to +\infty} \frac{1}{n} E[U^{(n)}] = 1 \tag{3.2}$$

Proof. For a fixed $\epsilon > 0$, observe that

$$\begin{aligned} P\left\{\left|\frac{U^{(n)}}{n} - 1\right| > \epsilon\right\} &= P\left\{\frac{U^{(n)}}{n} < 1 - \epsilon\right\} \\ &\leq \sum_{i=1}^{k} P\left\{\frac{U_{r_i}^{(n)}}{n} < 1 - \epsilon\right\} \\ &= \sum_{i=1}^{k} P\left\{\frac{U_{r_i}^{(n)}}{r_i} \leq \frac{[n(1-\epsilon)]}{r_i}\right\} \\ &\to 0 \quad as \quad n \to +\infty, \end{aligned} \tag{3.3}$$

where the first equality follows since at least one of the $U_{r_i}^{(n)}$'s is strictly less than n, and convergence to zero is established because

$$\lim_{n \to \infty} \frac{[n(1-\epsilon)]}{r_i} = \frac{1-\epsilon}{\pi_{i-1} - \pi_i} < \lim_{n \to +\infty} \frac{U_{r_i}^{(n)}}{r_i} = \frac{1}{\pi_{i-1} - \pi_i},$$

where the right-hand side limit is taken in the almost sure sense (A detailed proof of this convergence is provided in Appendix A), and hence, $\frac{U^{(n)}}{n} \to 1$ in probability as $n \to +\infty$. Convergence in mean then follows from (3.3) and the fact that $\frac{U^{(n)}}{n}$ is bounded by 1 with probability one. □

Theorem 4 presents the policy that achieves the optimal long-run expected reward per task.

Theorem 4. *A policy that assigns the j^{th} task to a type-i worker if $X_j \in (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$ and $j \leq U^{(n)}$ achieves the optimal long-run expected reward per task as given in* (3.1) *provided that F is continuous.*

Proof. To prove the result, the total reward obtained from assigning n jobs to n workers under this policy, R_n, is split into the reward obtained up to time $U^{(n)}$ and the reward obtained after $U^{(n)}$, which are denoted by $R_n^{(1)}$ and $R_n^{(2)}$, respectively. Observe that

$$\left|\frac{1}{n}E\left[R_n\right] - r^*\right| \leq \left|\frac{1}{n}E\left[R_n^{(1)}\right] - r^*\right| + \left|\frac{1}{n}E\left[R_n^{(2)}\right]\right|,$$

where

$$\begin{aligned} 0 \leq \left|\frac{1}{n}E\left[R_n^{(1)}\right] - r^*\right| &= \left|\frac{1}{n}E\left[\sum_{i=1}^{U^{(n)}}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}}\right] - r^*\right| \\ &\leq \sum_{j=1}^{k} p_j \left|E\left[\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_j\}}\right] - E\left[X_1 I_{\{X_1 \in \mathcal{A}_j\}}\right]\right| \\ &\leq p_1 \sum_{j=1}^{k} \left|E\left[\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_j\}}\right] - E\left[X_1 I_{\{X_1 \in \mathcal{A}_j\}}\right]\right|. \end{aligned} \tag{3.4}$$

Now, arbitrarily fix $1 \leq j^{'} \leq k$, and note that

$$\begin{aligned} \left|E\left[\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_{j'}\}}\right] - E\left[X_1 I_{\{X_1 \in \mathcal{A}_{j'}\}}\right]\right| \leq{}& \left|E\left[\frac{1}{n}\sum_{i=1}^{n} X_i I_{\{X_i \in \mathcal{A}_{j'}\}}\right] - E\left[X_1 I_{\{X_1 \in \mathcal{A}_{j'}\}}\right]\right| \\ &+ E\left[\frac{1}{n}\sum_{l=U^{(n)}+1}^{n} X_l I_{\{X_l \in \mathcal{A}_{j'}\}} I_{\{U^{(n)}<n\}}\right], \end{aligned} \tag{3.5}$$

since

$$\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_{j'}\}} = \frac{1}{n}\sum_{i=1}^{n} X_i I_{\{X_i \in \mathcal{A}_{j'}\}} - \left(\frac{1}{n}\sum_{l=U^{(n)}+1}^{n} X_l I_{\{X_l \in \mathcal{A}_{j'}\}}\right) I_{\{U^{(n)}<n\}}, \tag{3.6}$$

with probability one. The first term on the right-hand side of (3.5) is zero since task values are IID. Moreover,

$$\begin{aligned} E\left[\frac{1}{n}\left(\sum_{i=U^{(n)}+1}^{n} X_i I_{\{X_i \in \mathcal{A}_{j'}\}}\right) I_{\{U^{(n)}<n\}}\right] &\leq \frac{1}{n}E\left[\left(\sum_{i=U^{(n)}+1}^{n} X_i\right) I_{\{U^{(n)}<n\}}\right] \\ &= \frac{1}{n}E\left[\sum_{i=1}^{n} X_i - \sum_{j=1}^{U^{(n)}} X_j\right] \\ &= \left(1 - \frac{E\left[U^{(n)}\right]}{n}\right) E\left[X_1\right], \end{aligned}$$

where the first equality follows from the fact that $U^{(n)} \leq n$, and the second equality is obtained since $U^{(n)}$ is a stopping time with respect to the IID sequence $\{X_i\}$. Therefore, (3.4) is simplified to

$$0 \leq \left|\frac{1}{n}E\left[R_n^{(1)}\right] - r^*\right| \leq kp_1\left(1 - \frac{E\left[U^{(n)}\right]}{n}\right) E\left[X_1\right], \tag{3.7}$$

and hence, by Proposition 2, $\lim_{n\to+\infty} \frac{1}{n} E\left[R_n^{(1)}\right] = r^*$. Now, observe that

$$\begin{aligned} 0 \leq \frac{1}{n} E\left[R_n^{(2)}\right] &\leq \frac{1}{n} p_1 E\left[\left(\sum_{i=U^{(n)}+1}^{n} X_i\right) I_{\{U^{(n)}<n\}}\right] \\ &\leq p_1 \left(1 - \frac{E\left[U^{(n)}\right]}{n}\right) E\left[X_1\right] \\ &\to 0 \quad \text{as} \quad n \to +\infty, \end{aligned} \tag{3.8}$$

and hence, $\lim_{n\to\infty} \frac{1}{n} E\left[R_n\right] = r^*$. □

As it follows from the proof of Theorem 4, the way in which tasks are assigned to workers after time period $U^{(n)}$ has no effect in the long-run, since $\frac{E\left[R_n^{(2)}\right]}{n} \to 0$ as $n \to +\infty$. Theorem 5 and Corollary 2 discuss different convergence modes for the reward per task obtained under the optimal policy as $n \to +\infty$.

Theorem 5. *Suppose that the task values are bounded above, and let* $\sup \mathcal{S} := M$*, where* $\mathcal{S}$ *is the state space of task values. The reward per task obtained under the optimal policy, described in Theorem 4, converges in probability to* r^* *as* $n \to +\infty$*.*

Proof. The outline of the proof is along the same lines as that of Theorem 4, and hence, it has been omitted (See Appendix A for a detailed proof). □

Corollary 2. *Suppose that the task values are bounded above by* M*. The reward per task obtained under the optimal policy converges in the mean-square sense to* r^* *as* $n \to +\infty$*,*

$$\lim_{n\to+\infty} E\left[\left(\frac{R_n}{n} - r^*\right)^2\right] = 0.$$

Proof. Convergence in the mean-square sense follows from Theorem 5 and the fact that reward per task is bounded as follows:

$$0 \leq \frac{R_n}{n} \leq p_1 M,$$

for any $n = 1, 2, 3, \cdots$. □

Corollary 3 discusses the limiting behavior of the reward-per-task variance under the optimal policy as $n \to +\infty$.

Corollary 3. *With the boundedness assumption on task values (i.e.,* $\sup \mathcal{S} = M$*) and under the optimal policy proposed in Theorem 4, the long-run variance of reward per task is minimized and converges to zero as the number of tasks approaches infinity,*

$$\lim_{n\to+\infty} Var\left(\frac{R_n}{n}\right) = 0.$$

Proof. The result follows from the boundedness assumption and is omitted due to simplicity. □

Theorem 6 provides convergence rate results for the long-run expected reward per task under the optimal policy.

Theorem 6. *The expected reward per task under the optimal policy converges to r^* with an exponential rate as $n \to +\infty$.*

Proof. It follows from (3.7) and (3.8) that under the optimal policy

$$\left| \frac{1}{n} E[R_n] - \sum_{j=1}^{k} \left(p_j \int_{F^{-1}(\pi_{j-1})}^{F^{-1}(\pi_j)} xF(dx) \right) \right| \leq (k+1) p_1 E[X_1] \left| E\left[\frac{U^{(n)}}{n} - 1 \right] \right|,$$

and hence, to get the convergence rate of the long-run expected reward per task under the optimal policy, we need the rate at which $E\left[\frac{U^{(n)}}{n}\right]$ converges to 1. For any arbitrary $0 < \delta < 1$, it follows that

$$\begin{aligned}
\left| E\left[\frac{U^{(n)}}{n} - 1 \right] \right| &\leq E\left[\left| \frac{U^{(n)}}{n} - 1 \right| \right] \\
&\leq P\{ \left| \frac{U^{(n)}}{n} - 1 \right| > \delta \} + \delta \\
&= P\{ \frac{U^{(n)}}{n} < 1 - \delta \} + \delta \\
&= \sum_{i=1}^{k} P\{ \frac{U_{r_i}^{(n)}}{r_i} \leq \frac{[n(1-\delta)]}{r_i} \} + \delta.
\end{aligned}$$

Recall that $U_{r_i}^{(n)}$ is a negative binomial random variable with parameters $(r_i, \pi_{i-1} - \pi_i)$, and hence, it can be represented as the sum of r_i IID geometric random variables each having mean $\frac{1}{\pi_{i-1} - \pi_i}$. Now, let $a_i^{(n)} := \frac{[n(1-\delta)]}{r_i}$, and observe that $\lim_{n \to +\infty} a_i^{(n)} = \frac{1-\delta}{\pi_{i-1} - \pi_i} < \frac{1}{\pi_{i-1} - \pi_i}$. Therefore, the large deviation principle can be applied to obtain the following exponential convergence rate:

$$\left| E\left[\frac{U^{(n)}}{n} - 1 \right] \right| \leq \sum_{i=1}^{k} exp\{ -r_i I(a_i^{(n)}) \} + \delta,$$

where $I(a_i^{(n)})$ is a (convex) non-negative function. □

This section studied the limiting behavior of the SSAP with k worker categories where tasks are assumed to be IID following a known probability distribution. This assumption is relaxed in Section 3.2, where incomplete information (about the task distributions) and dependency between task values are incorporated into the model.

3.3 The Model: Unobservable Task Distributions

Consider the same SSAP with k (fixed) worker categories, where the i^{th} category consists of $r_i = [n\pi_{i-1}] - [n\pi_i]$ workers each with value p_i. Assume that the task values are random variables coming from r different distributions, where the successive distributions are governed by an irreducible ergodic time-homogeneous Markov chain with a known transition probability matrix $Q = (q_{ij})$ and an invariant distribution μ. Let Z_j denote the state of the Markov chain at time period j, with $\mathcal{S} = \{1, 2, \cdots, r\}$ being the state space of the Markov chain. More specifically, $Z_j = k$ means that the j^{th} task X_j is a random variable with distribution function F_k having support $\mathcal{B} \subseteq [0, +\infty)$. Assume that there exists a random variable Y that is independent of the task values and $P\{X_j \leq Y\} = 1$ for $j = 1, 2, \cdots$, where $E[Y] < +\infty$. Once a task arrives in a given time period, its value is observed; however, the state of the Markov chain (and hence, the distribution associated with the task value) is unobservable. The goal is to define an assignment policy that obtains the optimal expected reward per task after assigning all the tasks to workers as $n \to +\infty$. To this end, let $W = \{W_j, j = 1, 2, \cdots\}$ be a discrete-time Markov chain where $W_j := (Z_j, X_j)$, and note that only X_j is observable at time period j.

Note that due to the assumptions made above, task values are no longer IID, and hence, an approach different from that adopted in Section 3.2 must be applied. Moreover, as explained later in this section, this approach is distinct from that adopted by [4]. To solve the problem, first the chain W is proven to be positive recurrent so has an invariant distribution, which implies that the strong law of large numbers holds for W. From there, a stationary policy is defined to achieve the long-run optimal expected reward per task. Lemma 6 provides an invariant distribution $\bar{\mu}$ for W.

Lemma 6. *The chain W admits an invariant measure $\bar{\mu}$ where*

$$\bar{\mu}(k, E) := \mu(k) F_k(E), \tag{3.9}$$

for any $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$.

Proof. Let $P\{(l, x), (k, E)\}$ denote the probability of transitioning from state (l, x) to state (k, E). To prove the result, one needs to verify that for any $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$,

$$\bar{\mu}(k, E) = \int_{\mathcal{B}} \sum_{l=1}^{r} \bar{\mu}(l, dx) P\{(l, x), (k, E)\}, \tag{3.10}$$

where $\bar{\mu}$ is given by (3.9). To do so, (3.9) is substituted into the right-hand side of (3.10) as follows:

$$\begin{aligned}\int_{\mathcal{B}} \sum_{l=1}^{r} \bar{\mu}(l, dx) P\left\{(l, x), (k, E)\right\} &= \sum_{l=1}^{r} \int_{\mathcal{B}} \mu(l) F_l(dx) q_{lk} F_k(E) \\ &= F_k(E) \sum_{l=1}^{r} \mu(l) q_{lk} \int_{\mathcal{B}} F_l(dx) \\ &= F_k(E) \sum_{l=1}^{r} \mu(l) q_{lk} \\ &= F_k(E) \mu(k) \\ &= \bar{\mu}(k, E),\end{aligned}$$

where the equality before the last follows from the fact that μ is an invariant measure for $Z := \{Z_j, j = 1, 2, \cdots\}$.

□

Now, fix an arbitrary state $s_0 \in \mathcal{S}$, and define

$$\psi(k, E) := \delta_{s_0}(k) F_{s_0}(E), \tag{3.11}$$

for all $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$, where $\delta_{s_0}(k) := I_{\{k=s_0\}}$. Note that if $\psi(k, dx) > 0$, then $k = s_0$ and $F_k(dx) > 0$. Also, let $\tau_{(k,E)} = \min\{j \geq 1 : W_j \in (k, E)\}$, and define

$$\begin{aligned}L((l, x), (k, E)) : &= P_{(l,x)}\left\{\tau_{(k,E)} < +\infty\right\} \\ &= P\{\text{W ever enters } (k, E), \text{ starting from } (l, x)\},\end{aligned}$$

for $(l, x) \in \mathcal{S} \times \mathcal{B}$ and $(k, E) \subset \mathcal{S} \times \mathcal{B}$. Observe that $L((l, x), (k, E)) > 0$ implies that (k, E) is accesible from (l, x). Lemma 7 discusses a useful property of W.

Lemma 7. *The chain W is ψ-irreducible (see [20], Chapter 4, page 89) with ψ defined in* (3.11)*; i.e.,*

$$\text{if } \psi(k, E) > 0, \text{ then } L((l, x), (k, E)) > 0, \tag{3.12}$$

for any state $(l, x) \in \mathcal{S} \times \mathcal{B}$.

Proof. Fix $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$ such that $\psi(k, E) > 0$, and note that $k = s_0$ and $F_k(E) > 0$. Since Z is irreducible, there exists $n \geq 1$ such that $Q_{lk}^{(n)} := P\left\{Z_{t+n} = k \middle| Z_t = l\right\} > 0$, for $l \in \mathcal{S}$. Therefore, $P_{(l,x)}^{(n)}\{W \in (k, E)\} = Q_{lk}^{(n)} F_k(E) > 0$, for $x \in \mathcal{B}$, and hence, $L((l, x), (k, E)) > 0$. □

For Markov chains with general state space, this is basically the equivalent of the irreducibility concept. In fact, ψ-irreducibility ensures that large sets (i.e., sets with positive measure according to ψ) are eventually reached by the chain with positive probability, regardless of the starting point. Therefore, it guarantees

that the Markov chain will not split into separate parts. Verifying ψ-irreducibility is relatively easy but has many benefits. Lemma 8 follows from Proposition 4.2.2 in [20] and illustrates a direct consequence of ψ-irreducibility.

Lemma 8. *If W is ψ-irreducible for some measure ψ, then there exists a probability measure Ψ, called the maximal irreducibility measure, such that*

(1) *W is Ψ-irreducible.*

(2) *$\psi(A, E) = 0$ if $\Psi(A, E) = 0$, where $(A, E) \subset \mathcal{S} \times \mathcal{B}$.*

(3) *If $\Psi(A, E) = 0$ where $(A, E) \subset \mathcal{S} \times \mathcal{B}$, then $\Psi\{(l, x) : L((l, x), (A, E)) > 0\} = 0$.*

(4) *if $\Psi((A, E)^c) = 0$, then $(A, E) = (A_0, E_0) \cup (B, N)$ where $\Psi((B, N)) = 0$ and (A_0, E_0) is absorbing; i.e., $P\{(l, x), (A_0, E_0)\} = 1$ for all $(l, x) \in (A_0, E_0)$.*

Lemma 8 implies that Ψ-null sets are avoided by almost all points, and if we ignore these sets, we are left with an absorbing set. The chain W is called Ψ-irreducible if it is ψ-irreducible for some measure ψ and Ψ is the maximal irreducibility measure satisfying the conditions of Lemma 8 (see [20], Chapter 4).

Definition 1. The chain W is called Harris recurrent (see [20], Chapter 9, page 204) if it is Ψ-irreducible and

$$P_{(l,x)}\{W \in (A, E) \ \ i.o.\} = 1,$$

for all $(l, x) \in (A, E)$, where $(A, E) \subset \mathcal{S} \times \mathcal{B}$ and $\Psi(A, E) > 0$.

Definition 2. The chain W is called positive (see [20], Chapter 10, page 235) if it is Ψ-irreducible and admits an invariant probability measure.

Definition 3. If W is both Harris recurrent and positive, then it is called a positive Harris recurrent chain (see [20], Chapter 10, page 236).

A well-known and commonly-used concept in the theory of Markov chains on general state space is the property of *small set*, which is helpful is proving the existence of a stationary distribution, convergence rate to the stationary distribution, couplings, etc. Ψ-irreducibility along with notion of small sets develops the Markov chain theory on a general state space in complete analogy with the countable state space theory.

Definition 4. A set $(A, E) \subset \mathcal{S} \times \mathcal{B}$ is called a small set (see [20], Chapter 5, page 110) if there exists a positive integer m and a non-trivial measure ν_m, such that

$$P^{(m)}_{(l,x)}\{W \in (C, G)\} \geq \nu_m\{(C, G)\},$$

for all $(l, x) \in (A, E)$ and $(C, G) \subset \mathcal{S} \times \mathcal{B}$.

Assumption 1. There exists a small set (see [20], Chapter 5, page 109) $(\bar{A}, \bar{E}) \subset \mathcal{S} \times \mathcal{B}$ such that $L((l, x), (\bar{A}, \bar{E})) = 1$ for all $(l, x) \in \mathcal{S} \times \mathcal{B}$.

Corollary 4. *The chain W is positive Harris recurrent (see [20], Chapter 9, page 204, and Chapter 10, page 235) under Assumption* 1*.*

Proof. W is a positive chain since it is ψ-irreducible and admits an invariant probability measure $\bar{\mu}$ (see [20], Chapter 10). Moreover, Assumption 1 along with ψ-irreduciblity of W implies that the chain is Harris recurrent (Proposition 9.1.7 [20]). □

Corollary 5 proves that the strong law of large numbers holds true for the chain W, under Assumption 1, and it follows from Theorem 17.0.1 in [20].

Corollary 5. *For any function g defined on $\mathcal{S} \times \mathcal{B}$,*

$$\frac{1}{n}\sum_{j=1}^{n} g(W_j) \to \bar{\mu}(g) \quad \textit{almost surely as} \quad n \to +\infty,$$

for any g satisfying $\bar{\mu}(|g|) < +\infty$.

To present the optimal policy that achieves the maximum expected reward per task as $n \to +\infty$, some notation must be introduced first. Define $F : \mathcal{B} \to [0,1]$ as $F(a) := \sum_{j=1}^{r} \mu(j) F_j(a)$, and note that F is a distribution function on $\mathcal{B}$ with $F^{(-1)}(1) := +\infty$ and $F^{(-1)}(0) := -\infty$. A task X_j is labeled type-i if $X_j \in \mathcal{A}_i := (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$. For a fixed n, let $t_{r_i}^{(n)}$ denote the number of tasks that must arrive until r_i tasks of type i are obtained and define $t^{(n)} := \min\left\{t_{r_1}^{(n)}, \cdots, t_{r_k}^{(n)}\right\}$. Unless otherwise mentioned, Assumption 1 holds throughout Section 3.3. Lemma 9 discusses a useful property of t.

Lemma 9.

$$\lim_{n\to+\infty} \frac{E[\,t^{(n)}\,]}{n} = 1$$

Proof. Observe that $0 \leq \frac{t^{(n)}}{n} \leq 1$, and hence, to prove the lemma, it suffices to verify that

$$\frac{t^{(n)}}{n} \to 1 \quad \text{in probability as} \quad n \to +\infty.$$

For an arbitrarily fixed $\epsilon > 0$, observe that

$$\begin{aligned} P\left\{\left|\frac{t^{(n)}}{n} - 1\right| > \epsilon\right\} &= P\left\{t^{(n)} < n(1-\epsilon)\right\} \\ &= \sum_{i=1}^{k} P\left\{t_{r_i}^{(n)} \leq [n(1-\epsilon)]\right\} \\ &= \sum_{i=1}^{k} P\left\{\frac{\sum_{j=1}^{[n(1-\epsilon)]} I_{\{X_j \in \mathcal{A}_i\}}}{[n(1-\epsilon)]} \geq \frac{[n\pi_{i-1}] - [n\pi_i]}{[n(1-\epsilon)]}\right\}. \end{aligned} \tag{3.13}$$

Now, note that $I_{\{X_j \in \mathcal{A}_i\}} = \sum_{u=1}^{r} I_{\{X_j \in \mathcal{A}_i, Z_j = u\}}$, and obtain the steady-state mean of $I_{\{X_j \in \mathcal{A}_i\}}$ as follows:

$$\begin{aligned}
\bar{\mu}\left(I_{\{X_j \in \mathcal{A}_i\}}\right) &= \sum_{u=1}^{r} \bar{\mu}(u, \mathcal{A}_i) \\
&= \sum_{u=1}^{r} \mu(u) \left[F_u\left(F^{-1}(\pi_{i-1})\right) - F_u\left(F^{-1}(\pi_i)\right)\right] \\
&= F(F^{-1}(\pi_{i-1})) - F(F^{-1}(\pi_{i-1})) \\
&= \pi_{i-1} - \pi_i.
\end{aligned}$$

This implies that

$$\frac{\sum_{j=1}^{[n(1-\epsilon)]} I_{\{X_j \in \mathcal{A}_i\}}}{[n(1-\epsilon)]} \to \pi_{i-1} - \pi_i \quad \text{almost surely as} \quad n \to +\infty,$$

by Corollary 5, while

$$\lim_{n \to +\infty} \frac{[n\pi_{i-1}] - [n\pi_i]}{[n(1-\epsilon)]} = \frac{\pi_{i-1} - \pi_i}{1-\epsilon} > \pi_{i-1} - \pi_i,$$

and hence, $P\left\{\left|\frac{t^{(n)}}{n} - 1\right| > \epsilon\right\} \to 0 \quad \text{as} \quad n \to +\infty.$ □

To compute the expected reward per task in the long-run and to prove the main result, [4] uses Wald's equation in Lemma 2; however, this approach is not applicable in our problem, since $t^{(n)}$ is not a stopping time with respect to the IID sequence of tasks derived from a given fixed distribution. Let $\bar{\phi}$ be a policy that assigns X_j to p_i if $X_j \in \mathcal{A}_i$ and $j \leq t^{(n)}$. Lemma 10 computes the long-run expected reward per task until time period $t^{(n)}$ under policy $\bar{\phi}$.

Lemma 10.

$$\lim_{n \to +\infty} E\left[\frac{\sum_{i=1}^{k} p_i \sum_{j=1}^{t^{(n)}} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n}\right] = \sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx)$$

Proof. Note that

$$\frac{\sum_{j=1}^{t^{(n)}} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} = \frac{\sum_{j=1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} - \frac{\sum_{j=t^{(n)}+1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} I_{\{t^{(n)} < n\}},$$

where

$$\frac{\sum_{j=1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} \to \int_{\mathcal{A}_i} xF(dx) \quad \text{almost surely as} \quad n \to +\infty, \tag{3.14}$$

by Corollary 5. Moreover, $\frac{\sum_{j=1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} \leq Y$ with probability one, where $E[Y] < +\infty$. Therefore,

$$E\left[\frac{\sum_{j=1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n}\right] \to \int_{\mathcal{A}_i} xF(dx) \quad as \quad n \to +\infty, \tag{3.15}$$

by (3.14). Now, observe that

$$
\begin{aligned}
E\left[\frac{\sum_{j=t^{(n)}+1}^{n} X_j I_{\{X_j \in \mathcal{A}_i\}}}{n} I_{\{t^{(n)}<n\}}\right] &\leq E\left[Y\left(1-\frac{t^{(n)}}{n}\right) I_{\{t^{(n)}<n\}}\right] \\
&= E\left[Y\left(1-\frac{t^{(n)}}{n}\right)\right] \\
&= E[Y]\, E\left[1-\frac{t^{(n)}}{n}\right] \to 0 \quad as \quad n \to +\infty,
\end{aligned} \tag{3.16}
$$

where the last equality follows from the assumption that Y is independent from the task values, and convergence to zero is established by Lemma 9. □

Lemma 11 provides a lower bound for the optimal expected reward per task in the long-run.

Lemma 11.

$$
\sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx) \leq \liminf_{n\to+\infty} \frac{E[R_n^*]}{n},
$$

where R_n^* is the optimal total reward obtained after assigning all n tasks to the n workers.

Proof. Let $R_n^{\bar{\phi}}$ denote the total reward under policy $\bar{\phi}$. It follows that $R_n^{\bar{\phi}} = R_n^{\bar{\phi},1} + R_n^{\bar{\phi},2}$, where $R_n^{\bar{\phi},1}$ and $R_n^{\bar{\phi},2}$ denote the total reward obtained up to time period $t^{(n)}$ and after time period $t^{(n)}$, respectively. Recall that by Lemma 10,

$$
\lim_{n\to+\infty} \frac{E\left[R_n^{\bar{\phi},1}\right]}{n} = \sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx).
$$

For the total reward obtained after time period t, observe that from (3.16),

$$
\frac{E\left[R_n^{\bar{\phi},2}\right]}{n} \leq p_1 E\left[\frac{\sum_{j=t^{(n)}+1}^{n} X_j}{n} I_{\{t^{(n)}<n\}}\right] \to 0 \quad as \quad n \to +\infty,
$$

and hence,

$$
\lim_{n\to+\infty} \frac{E\left[R_n^{\bar{\phi}}\right]}{n} = \sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx).
$$

The result then follows since $\bar{\phi}$ is an arbitrary feasible policy and $\frac{E[R_n^{\bar{\phi}}]}{n} \leq \frac{E[R_n^*]}{n}$ for all n. □

Note that the proof of Lemma 11 implies that the way in which tasks are assigned to workers after time period $t^{(n)}$ has no effect in the long-run, since $\frac{E[R_n^{\phi,2}]}{n} \to 0$ as $n \to +\infty$ for any arbitrary policy ϕ.

Lemma 12 provides the optimal policy and the optimal expected reward per task for a problem with $k = 2$ worker categories as $n \to +\infty$.

Lemma 12. *Consider the special case of $k = 2$ with $r_1 = n - [n\pi]$, $r_2 = [n\pi]$, and $(p_1, p_2) = (1, 0)$. $\bar{\phi}$ assigns a task with value x to a worker with value one if $x \geq F^{-1}(\pi)$ and achieves the optimal expected reward in the long-run, where*

$$
\lim_{n\to+\infty} \frac{E[R_n^*]}{n} = \int_{F^{-1}(\pi)}^{+\infty} xF(dx).
$$

Proof. Observe that by Lemma 11,

$$\int_{F^{-1}(\pi)}^{+\infty} xF(dx) \leq \liminf_{n\to+\infty} \frac{E\left[R_n^*\right]}{n},$$

and hence, it remains to show that

$$\limsup_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} \leq \int_{F^{-1}(\pi)}^{+\infty} xF(dx).$$

To this end, let $X_{(j)}$ denote the j^{th} order statistic of $X_1, X_2, \cdots, X_n$ and note that no policy can do better than the policy that assigns $X_{([n\pi]+1)}, X_{([n\pi]+2)}, \cdots, X_{(n)}$ to workers with value one (since these tasks have the $n - [n\pi]$ highest values among all the arriving tasks). This implies that

$$\limsup_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} \leq \limsup_{n\to+\infty} \frac{E\left[\sum_{j=[n\pi]+1}^{n} X_{(j)}\right]}{n}, \tag{3.17}$$

where

$$\limsup_{n\to+\infty} E\left[\frac{\sum_{j=[n\pi]+1}^{n} X_{(j)}}{n}\right] \leq \int_{F^{-1}(\pi)}^{+\infty} xF(dx), \tag{3.18}$$

which follows from an approach similar to Lemma 4 in [4]. □

Corollary 6 generalizes the result of Corollary 2 in [7] that was originally proven for IID sequences of task values with a known observable probability distribution.

Corollary 6. *Let $X_{(j)}$ denote the j^{th} order statistic of tasks $X_1, X_2, \cdots, X_n$, coming from r different distributions $\{F_1, F_2, \cdots, F_r\}$, where the successive distributions are unobservable and are governed by an irreducible ergodic Markov chain with invariant distribution μ. It follows that*

$$\lim_{n\to+\infty} E\left[\frac{\sum_{j=[n\pi]+1}^{n} X_{(j)}}{n}\right] = \int_{F^{-1}(\pi)}^{+\infty} xF(dx),$$

and

$$\lim_{n\to+\infty} E\left[\frac{\sum_{j=1}^{[n\pi]} X_{(j)}}{n}\right] = \int_{-\infty}^{F^{-1}(\pi)} xF(dx).$$

Proof. Observe that Lemma 12, (3.17), and (3.18) imply that

$$\limsup_{n\to+\infty} E\left[\frac{\sum_{j=[n\pi]+1}^{n} X_{(j)}}{n}\right] = \int_{F^{-1}(\pi)}^{+\infty} xF(dx). \tag{3.19}$$

On the other hand,

$$\int_{-\infty}^{F^{-1}(\pi)} xF(dx) = \liminf_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} \leq \liminf_{n\to+\infty} \frac{E\left[\sum_{j=[n\pi]+1}^{n} X_{(j)}\right]}{n},$$

and hence,

$$\lim_{n\to+\infty} E\left[\frac{\sum_{j=[n\pi]+1}^{n} X_{(j)}}{n}\right] = \int_{F^{-1}(\pi)}^{+\infty} xF(dx).$$

□

Theorem 7 provides the optimal policy and the optimal long-run expected reward per task for a SSAP with k worker categories and unobservable task distributions.

Theorem 7. *For a SSAP with k worker categories, $\bar{\phi}$ achieves the optimal expected reward per task in the long-run, with*

$$\lim_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} = \sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx).$$

Proof. Consider a policy ϕ_{max} that assigns $X_{([n\pi_l]+1)}, X_{([n\pi_l]+2)}, \cdots, X_{([n\pi_{l-1}])}$ to type-l workers (i.e., with value p_l) for $l = 1, 2, \cdots, k$. Since no policy can do better than this (infeasible) policy, it follows that

$$\limsup_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} \leq \limsup_{n\to+\infty} \frac{E\left[R_n^{\phi_{max}}\right]}{n},$$

where

$$\frac{E\left[R_n^{\phi_{max}}\right]}{n} = p_k E\left[\frac{\sum_{j=1}^{[n\pi_{k-1}]} X_{(j)}}{n}\right] + p_{k-1} E\left[\frac{\sum_{j=[n\pi_{k-1}]+1}^{[n\pi_{k-2}]} X_{(j)}}{n}\right] + \cdots + p_1 E\left[\frac{\sum_{j=[n\pi_1]+1}^{n} X_{(j)}}{n}\right],$$

which along with Corollary 6 implies that

$$\limsup_{n\to+\infty} \frac{E\left[R_n^*\right]}{n} \leq \sum_{i=1}^{k} p_i \int_{F^{(-1)}(\pi_i)}^{F^{(-1)}(\pi_{i-1})} xF(dx) = \sum_{i=1}^{k} p_i \int_{\mathcal{A}_i} xF(dx).$$

Lemma 11 then completes the proof. □

Chapter 4

Limiting Behavior of the Target-dependent Stochastic Sequential Assignment Problem

4.1 Introduction

Consider the sequential stochastic assignment problem (SSAP) introduced in [12], where n workers are available to perform n IID sequentially-arriving tasks. The random variable X_j denotes the j^{th} task value, and a value (success rate) p_i is associated with each worker. Whenever the i^{th} worker is assigned to the j^{th} task, that worker becomes unavailable for future assignments, with $p_i x_j$ denoting the expected reward due to this assignment. The objective is to assign these n workers to n tasks so as to maximize the expected total reward. It is shown in [12] that there exists numbers

$$-\infty = a_{0,n} \leq a_{1,n} \leq a_{2,n} \leq \cdots \leq a_{n,n} = +\infty,$$

such that the optimal choice in the initial stage is to assign the i^{th} best available worker if the random variable X_1 falls within the i^{th} highest interval. The SSAP has applications in several areas, and various extensions to the problem have been discussed in the literature. For example, [23] studies a variation of the SSAP in aviation security screening systems, while [31] addresses the problem of allocating sequentially-arriving kidneys to patients on a waiting list. Another application of the SSAP is the asset selling problem [5], where one needs to choose the best offers out of a sequence of bids from potential buyers. Moreover, Albright [2] studies the SSAP with various task-arrival-time distributions. Nikolaev and Jacobson [22] consider a variation of SSAP in which the number of tasks is unknown until after the final arrival and follows a given probability distribution.

Implementing the optimal assignment policy for the SSAP, as described in [12], involves calculating a new set of breakpoints upon the arrival of each task. The computation of these breakpoints takes polynomial time but is cumbersome for large-scale problems; for example, consider a SSAP which allows passengers to be assigned to available aviation security resources, based on their perceived threat levels, as they check in at an airport [9]. Assigning passengers to security devices by re-calculating the breakpoints every time a passenger arrives is not practical, even for a small airport. Therefore, this work focuses on the limiting behavior of the $\{a_{i,n}\}$, as n approaches infinity, so as to obtain simpler solutions that are implementable in real-world problems. On the other hand, the existing SSAP literature focuses on a risk-neutral objective

function, seeking an assignment policy that maximizes the expected total reward. However, a risk-neutral policy is not always desirable since the probability distribution function (*pdf*) of the total reward may carry with it a high probability of low unaccepted values; therefore, there are instances that a decision maker is interested in a stable reward and looks for a risk-sensitive optimal assignment policy.

Taking the above-mentioned issues into consideration, this work studies the limiting behavior of the SSAP under a different objective function, called the *threshold criterion*. For a given *threshold* (or target) τ, the goal is to find a policy ϕ^* that minimizes the *threshold probability*: the probability (or risk) of the long-run reward per task failing to achieve the target τ. Specifically, the threshold critetion can be mathematically expressed as

$$\inf_{\phi\in\Phi} P\left\{\limsup_{n\to+\infty}\frac{1}{n}R_n^{\phi}\leq\tau\right\},$$

where Φ is the set of all admissible policies and R_n^{ϕ} is the total reward obtained after assigning all n tasks under policy ϕ. For simplicity, this problem is denoted as the LTSSAP since it studies the limiting behavior of the target-dependent SSAP. Two versions of this problem are studied here. The first version assumes that the sequentially-arriving tasks are IID with a known distribution function. The second problem then considers the case that the task values are derived from r different distributions, where the successive distributions are governed by an ergodic Markov chain. Once a task arrives in a given time period, its value is observed; however, the distribution it comes from is unobservable. In both problems, it is assumed that there exist k worker categories, where the ith category consists of r_i workers each with value p_i such that $\sum_{i=1}^{k} r_i = n$. Stationary policies are presented for both problems, which apart from minimizing the threshold probability, achieve the optimal long-run expected reward per task. These policies consist of $k-1$ fixed (time-independent) breakpoints, as opposed to the policy described by [12] in which the number of breakpoints increases with n and the (time-dependent) breakpoints are recalculated each time a task arrives.

In the existing literature, [7] and [4] address the limiting behavior of a special case of SSAP (called the secretary problem) where $p_i \in \{0,1\}$. [4] assumes that the task values are generated by r different distributions, and once a task arrives, both its value and its distribution are observed. However, the assumption that the underlying task distributions are observable does not hold in most real-world problems. Deviating from the existing literature, the present work considers the task distributions to be unobservable and forming a hidden Markov chain. As an application of the problem with a hidden Markov chain, consider the case where task values represent the worth of arriving tasks, where the task worth is dependent on the economic conditions upon the arrival of that task ([21]). Clearly, the conditions of economy vary from time to time, and this can be captured by our model through the Markov chain, where the states of the chain correspond to the economic conditions. Although [7] has laid out the framework for studying the limiting behavior of the SSAP, this work takes on a different proof approach, mainly since (1) dealing with the LTSSAP involves

studying the almost sure convergence of the long-run reward per task, while the existing literature focuses on the convergence of the long-run expected value of reward per task and (2) as mentioned in [8], the generalization of the problem with $p_i \in \{0, 1\}$ to the case with arbitrary worker values is not possible using the approach applied in [7].

Consider an application of LTSSAP in aviation security, where sequentially-arriving passengers are assigned to available security resources, as they check in at an airport. A random variable X_j is associated with passenger j, denoting their threat (risk) value. Threat value is defined as the probability of a passenger carrying a threat item. Once a passenger arrives, a prescreening system determines their threat value, and assigns them to either a non-selectee class (i.e., a class of passengers who have been cleared of posing a threat to the airport) or a selectee class (i.e., the class who have not been cleared). A security level is assigned to each class, denoting the probability of detecting a passenger with a threat item. Let L_S and L_{NS} be the security levels associated with the selectee and the non-selectee classes. Moreover, let $\gamma_j = 1$ and $\gamma_j = 0$ denote the j^{th} passenger assignment as a selectee and a non-selectee, respectively. The *total security* for this setting is defined as

$$\sum_{j=1}^{n} X_j \left[L_S \gamma_j + L_{NS}(1 - \gamma_j) \right].$$

At any airport, it is critical to maintain a stable and reasonable level of security with high probability, at all times. Therefore, the objective is to find a policy for assigning passengers to classes as they check in so as to minimize the probability of the long-run average security failing to achieve the target τ.

The chapter is organized as follows: Section 4.2 provides an optimal assignment policy for LTSSAP with IID task values from a known observable distribution. The problem of unobservable task distributions, forming a hidden Markov chain, is addressed in Section 4.3. Section 4.4 presents concluding remarks and future research directions.

4.2 The Model: Observable Task Distributions

Consider the SSAP with n tasks and k (fixed) worker categories, where the i^{th} worker-category consists of r_i workers each with value p_i. Let $[.]$ denote the floor function where $[y] := \max\left\{m \in \mathbb{Z} \middle| m \leq y\right\}$. Moreover, let π_i be the fraction of total number of workers that belong to categories $i+1$ to k, and hence, $\alpha_i := \pi_{i-1} - \pi_i$ denotes the fraction of workers assigned to class i, for $i = 1, 2, \cdots, k$. For simplicity, the i^{th} worker category is referred to as type-i workers, with the size of the i^{th} category given by

$$r_i = [n\pi_{i-1}] - [n\pi_i] \quad \text{for } i = 1, 2, \cdots, k,$$

where $\pi_0 = 1$, $\pi_k = 0$, and $\pi_{i+1} < \pi_i$ for $i = 0, 1, \cdots, k-1$. Also, assume that $p_{i+1} < p_i$ for $i = 1, 2, \cdots, k$.

Let $\mathcal{A}_i := (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$ for $i = 1, 2, \cdots, k$, where F is the distribution function of task values. A task X_j is labeled type-i if $X_j \in \mathcal{A}_i$. Consider a policy ϕ_L that assigns the j^{th} task to a type-i worker if $X_j \in \mathcal{A}_i$. If the goal is to maximize the expected reward per task as $n \to +\infty$, then it can be deduced from [7] that the optimal long-run expected reward per task is given by

$$r^* := \sum_{j=1}^{k} \left(p_j \int_{F^{-1}(\pi_j)}^{F^{-1}(\pi_{j-1})} xF(dx) \right), \tag{4.1}$$

with $F^{-1}(1) = +\infty$ and $F^{-1}(0) = -\infty$. Moreover, [8] proves that ϕ_L is the stationary policy that achieves the optimal long-run expected reward per task, r^*, for this SSAP. In what follows, it is proven that this policy also optimizes the threshold probability for a given target value τ; specifically, it solves the LTSSAP and achieves the infimum in the following expression

$$\inf_{\phi \in \Phi} P\left\{ \limsup_{n \to +\infty} \frac{1}{n} R_n^{\phi} \leq \tau \right\}, \tag{4.2}$$

where R_n^{ϕ} is the total reward obtained after assigning all n tasks under policy ϕ. Note that this policy consists of $k-1$ (time-independent) fixed breakpoints, where k does not change as $n \to +\infty$, as opposed to the policy described by [12] in which the number of breakpoints increases with n and the breakpoints are recalculated at each time period.

For a fixed n, let $U_{r_i}^{(n)}$ denote the number of tasks that arrive until r_i tasks of type i are obtained and define $U^{(n)} := \min\left\{ U_{r_1}^{(n)}, U_{r_2}^{(n)}, \cdots, U_{r_k}^{(n)} \right\}$. Observe that $U_{r_i}^{(n)}$ follows a negative binomial distribution with parameters $(r_i, \pi_{i-1} - \pi_i)$, since tasks are assumed to be IID. Proposition 3 presents a useful property of $U^{(n)}$.

Proposition 3.

$$\frac{U^{(n)}}{n} \to 1 \quad \textit{almost surely} \quad \textit{as} \quad n \to +\infty \tag{4.3}$$

Proof. Fix $i \in \{1, 2, \cdots, k\}$, and recall that $U_{r_i}^{(n)}$ is a negative binomial random variable with parameters $(r_i, \pi_{i-1} - \pi_i)$, and hence, it can be represented as the sum of r_i IID geometric random variables each having mean $\frac{1}{\pi_{i-1} - \pi_i}$. The strong law of large numbers (SLLN) implies that

$$\frac{U_{r_i}^{(n)}}{r_i} \to \frac{1}{\pi_{i-1} - \pi_i} \quad \text{as} \quad n \to +\infty,$$

almost surely, which leads to

$$\frac{U_{r_i}^{(n)}}{n} = \frac{U_{r_i}^{(n)}}{r_i} . \frac{r_i}{n} \quad \to 1 \quad \text{as} \quad n \to +\infty,$$

almost surely. Recall that the minimum of any two arbitrary functions f and g can be represented as

$$\min\{f,g\} = \frac{1}{2}\left(f + g - |f-g|\right),$$

and hence,

$$\frac{U^{(n)}}{n} \to 1 \quad \text{as} \quad n \to +\infty,$$

almost surely, since it is the minimum over a finite number of almost-surely convergent functions. □

Applying the result in Proposition 3, Theorem 8 proves the optimality of ϕ_L for $\tau \in [-\infty, r^*)$.

Theorem 8. *Assume that $\tau < r^*$. The infimum in (4.2) is achieved by a policy ϕ_L that assigns the j^{th} task to a type-i worker if $X_j \in \mathcal{A}_i$ and $j \leq U^{(n)}$.*

Proof. To prove the result, the total reward under ϕ_L, $R_n^{\phi_L}$, is split into the reward obtained up to time $U^{(n)}$ and the reward obtained after $U^{(n)}$, which are denoted by $R_n^{(1)}$ and $R_n^{(2)}$, respectively. The superscript ϕ_L is dropped to simplify the notation. Observe that

$$\begin{aligned}\frac{1}{n}R_n^{(1)} &= \frac{1}{n}\sum_{i=1}^{U^{(n)}}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}} \\ &= \frac{U^{(n)}}{n}\cdot\frac{1}{U^{(n)}}\sum_{i=1}^{U^{(n)}}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}},\end{aligned}$$

and note that

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}} \to r^* \quad \text{as} \quad n \to +\infty, \tag{4.4}$$

almost surely by the SSLN. Moreover, $U^{(n)} \geq \min\{r_1, r_2, \cdots, r_k\}$, and hence, $U^{(n)} \to +\infty$ almost surely as $n \to +\infty$. This fact combined with (4.4) results in

$$\frac{1}{U^{(n)}}\sum_{i=1}^{U^{(n)}}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}} \to r^* \quad \text{as} \quad n \to +\infty, \tag{4.5}$$

almost surely. Therefore, $\frac{1}{n}R_n^{(1)} \to r^*$ almost surely as $n \to +\infty$, by (4.5) and Proposition 3. On the other hand,

$$\begin{aligned}0 \leq \frac{1}{n}R_n^{(2)} &\leq p_1\left(\frac{1}{n}\sum_{i=U^{(n)}+1}^{n} X_i\right) I_{\{U^{(n)}<n\}} \\ &= p_1\left(\frac{1}{n}\sum_{i=1}^{n} X_i - \frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i\right) \to 0 \quad \text{as} \quad n \to +\infty,\end{aligned}$$

almost surely, by a similar argument. Therefore, $\frac{1}{n}R_n \to r^*$ almost surely as $n \to +\infty$ under ϕ_L, and it follows that

$$P_{\phi_L}\left\{\limsup_{n\to+\infty}\frac{1}{n}R_n \leq \tau\right\} = 0,$$

since $\tau < r^*$, implying that ϕ_L achieves the infimum in (4.2). $\square$

Note that Theorem 8 along with the results in [8], imply that the policy ϕ_L achieves the maximum long-run expected reward per task, while minimizing the risk of the long-run reward per task failing to achieve a given target level. Theorem 9 proves a result which is useful in solving the LTSSAP for target values greater than r^*.

Assumption 2. There exists a random variable Y that is independent of the task values and $P\{X_j \leq Y\} = 1$ for $j = 1, 2, \cdots$, where $E[Y] < +\infty$.

Theorem 9. *Consider a policy ϕ_B that assigns $X_{(j)}$ (the j^{th} order statistic of $X_1, X_2, \cdots, X_n$) to the j^{th} best available worker. Under Assumption 2, it follows that*

$$\frac{1}{n} R_n^{\phi_B} \to r^* \quad \textit{almost surely} \quad \textit{as} \quad n \to +\infty. \tag{4.6}$$

Proof. Note that the reward per task under ϕ_B can be expressed as

$$\frac{1}{n} R_n^{\phi_B} = \sum_{j=1}^{k} p_j \left(\frac{1}{n} \sum_{i=[n\pi_j]+1}^{[n\pi_{j-1}]} X_{(i)} \right),$$

while $r^* = \sum_{j=1}^{k} p_j E\left[X_1 I_{\{X_1 \in \mathcal{A}_j\}}\right]$. Therefore, to prove (4.6), it suffices to prove

$$\left(\frac{1}{n} \sum_{i=[n\pi_j]+1}^{[n\pi_{j-1}]} X_{(i)} \right) \to E\left[X_1 I_{\{X_1 \in \mathcal{A}_j\}}\right] \quad \text{as} \quad n \to +\infty,$$

almost surely for $j = 1, 2, \cdots, k$. To this end, one can alternatively prove that

$$\frac{1}{n} \sum_{i=[n\pi]+1}^{n} X_{(i)} \to E\left[X_1 I_{\{X_1 \in \mathcal{A}_1\}}\right] \quad \text{as} \quad n \to +\infty,$$

almost surely, for arbitrarily fixed $0 < \pi < 1$. Observe that $\mathcal{A}_1 = [F^{-1}(\pi), +\infty)$ and

$$\frac{1}{n} \sum_{i=1}^{n} X_i I_{\{X_i \geq F^{-1}(\pi)\}} \to E\left[X_1 I_{\{X_1 \geq F^{-1}(\pi)\}}\right] \quad \text{as} \quad n \to +\infty,$$

almost surely by SLLN, and hence, it remains to show that

$$\frac{1}{n} \Big| \sum_{i=[n\pi]+1}^{n} X_{(i)} - \sum_{j=1}^{n} X_j I_{\{X_j \geq F^{-1}(\pi)\}} \Big| \to 0 \quad \text{as} \quad n \to +\infty, \tag{4.7}$$

almost surely. Define $N(n) := \sum_{i=1}^{n} I_{\{X_i \geq F^{-1}(\pi)\}}$, and note that $\frac{1}{n} N(n) \to 1 - \pi$ almost surely as $n \to +\infty$. Consider the following two cases: (1) if $N(n) \geq n - [n\pi]$, then the left-hand numerator in (4.7) contains $N(n) - n + [n\pi]$ terms, all less than or equal to Y and (2) if $N(n) < n - [n\pi]$, then the left-hand numerator

in (4.7) contains $n - [n\pi] - N(n)$ terms, each less than $F^{-1}(\pi)$. Therefore,

$$\frac{1}{n}\Big|\sum_{i=[n\pi]+1}^{n} X_{(i)} - \sum_{j=1}^{n} X_j I_{\{X_j \geq F^{-1}(\pi)\}}\Big| \leq \Big|\frac{n-[n\pi]-N(n)}{n}\Big|\left(Y \vee F^{-1}(\pi)\right) \to 0 \quad \text{as} \quad n \to +\infty,$$

almost surely by SLLN and the fact that $Y < +\infty$ with probability one. □

Using the result from Theorem 9, Corollary 7 solves (4.2) for $\tau \in [r^*, +\infty)$.

Corollary 7. *If $\tau \geq r^*$, then*

$$\inf_{\phi \in \Phi} P\left\{\limsup_{n\to+\infty} \frac{1}{n} R_n^{\phi} \leq \tau\right\} = 1. \tag{4.8}$$

Proof. Observe that

$$P_{\phi_B}\left\{\limsup_{n\to+\infty} \frac{1}{n} R_n \leq \tau\right\} = 1,$$

by Theorem 9. Also,

$$P_{\phi_B}\left\{\limsup_{n\to+\infty} \frac{1}{n} R_n \leq \tau\right\} \leq P_{\phi}\left\{\limsup_{n\to+\infty} \frac{1}{n} R_n \leq \tau\right\},$$

for any $\phi \in \Phi$, since no admissible policy can outperform ϕ_B, and hence, (4.8) follows. □

By Corollary 7, if the target value is greater than or equal to r^*, then all the admissible policies perform the same in terms of minimizing the threshold probability, and hence, the decision maker is indifferent between any two such policies. If the decision maker intends to optimize the assignments so as to achieve the maximum long-run expected reward per task, along with controling risk level in the sense of (4.2), then they can opt to apply policy ϕ_L.

4.3 The Model: Unobservable Task Distributions

Consider the LTSSAP with k worker categories, where the i^{th} category consists of $r_i = [n\pi_{i-1}] - [n\pi_i]$ workers, each with value p_i. Assume that the task values are generated from r different distributions, where the successive distributions are governed by an irreducible ergodic time-homogeneous Markov chain with (known) transition probability matrix $Q = (q_{ij})$ and an invariant (stationary) distribution μ. The state of the Markov chain at time period j is denoted by Z_j, with $\mathcal{S} = \{1, 2, \cdots, r\}$ being the state space of the Markov chain. Specifically, $Z_j = k$ means that the j^{th} task X_j is a random variable with distribution function F_k having support $\mathcal{B} \subseteq [0, +\infty)$. Upon the arrival of each task, its value is observed; however, the state of the Markov chain (and hence, the distribution associated with the task value) is unobservable. The goal is to come up with an assignment policy that minimizes the threshold probability in (4.2).

Since task values are derived from r distinct distributions that are linked together through a Markov chain, it follows that task values are no longer IID, and hence, the approach presented in Section 4.2 can't be used. To solve this problem, let $W = \{W_j, j = 1, 2, \cdots\}$ be a discrete-time Markov chain with state space

$\mathcal{S}\times\mathcal{B}$ where $W_j := (Z_j, X_j)$, and note that only X_j is observable at time period j. In what follows, the chain W is proven to be positive recurrent, implying that the strong law of large numbers holds for W. Then, it is shown that a stationary policy similar to ϕ_L achieves the infimum in (4.2) and is optimal. Lemma 13 provides an invariant distribution $\bar{\mu}$ for W.

Lemma 13. *The chain W admits an invariant measure $\bar{\mu}$ where*

$$\bar{\mu}(k, E) := \mu(k)F_k(E), \tag{4.9}$$

for any $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$.

Proof. Let $P\{(l, x), (k, E)\}$ denote the probability of transitioning from state (l, x) to state (k, E), where $(l, x) \in \mathcal{S} \times \mathcal{B}$. To prove the result, one needs to verify that for any $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$,

$$\bar{\mu}(k, E) = \int_{\mathcal{B}} \sum_{l=1}^{r} P\{(l, x), (k, E)\}\, \bar{\mu}(l, dx), \tag{4.10}$$

where $\bar{\mu}$ is given by (4.9). To do so, (4.9) is substituted into the right-hand side of (4.10) as follows:

$$\begin{aligned}
\int_{\mathcal{B}} \sum_{l=1}^{r} \bar{\mu}(l, dx) P\{(l, x), (k, E)\} &= \sum_{l=1}^{r} \left[\mu(l) \left(\int_{\mathcal{B}} F_l(dx)\right) q_{lk} F_k(E)\right] \\
&= F_k(E) \sum_{l=1}^{r} \mu(l) q_{lk} \\
&= \bar{\mu}(k, E),
\end{aligned}$$

where the last equality follows from the fact that μ is an invariant measure for $Z := \{Z_j, j = 1, 2, \cdots\}$, and (4.10) is verified. □

Before proceeding to Corollay 9, which indicates that the SLLN holds for the chain W (and hence, simplifies the proof of the desired results for the unobservable distributions case), some notation and definitions must be introduced, and auxiliary results must be presented.

Definition 5. Let $\tau_{(k,E)} := \min\{j \geq 1 : W_j \in (k, E)\}$ denote the first hitting of the set (k, E), and define

$$\begin{aligned}
L((l, x), (k, E)) : &= P_{(l,x)}\left\{\tau_{(k,E)} < +\infty\right\} \\
&= P\{\text{W ever enters } (k, E), \text{ starting from } (l, x)\},
\end{aligned}$$

for $(l, x) \in \mathcal{S} \times \mathcal{B}$ and $(k, E) \subset \mathcal{S} \times \mathcal{B}$.

Definition 6. The chain W is ψ-irreducible (see [20], Chapter 4, page 89) if there exists a measure ψ such that

$$\textit{if } \psi(k, E) > 0, \textit{ then } L((l, x), (k, E)) > 0, \tag{4.11}$$

for any state $(l, x) \in \mathcal{S} \times \mathcal{B}$.

Lemma 14. *Fix an arbitrary state $s_0 \in \mathcal{S}$, and define*

$$\psi(k, E) := \delta_{s_0}(k) F_{s_0}(E), \tag{4.12}$$

for all $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$, where $\delta_{s_0}(k) := I_{\{k=s_0\}}$. The chain W is ψ-irreducible with ψ defined in (4.12).

Proof. Fix $k \in \mathcal{S}$ and $E \subseteq \mathcal{B}$ such that $\psi(k, E) > 0$, and note that $k = s_0$ and $F_k(E) > 0$. Since Z is irreducible, there exists $n \geq 1$ such that $Q_{lk}^{(n)} := P\left\{Z_{t+n} = k \middle| Z_t = l\right\} > 0$, for any $l \in \mathcal{S}$. Therefore, $P_{(l,x)}^{(n)}\{W \in (k, E)\} = Q_{lk}^{(n)} F_k(E) > 0$, for $x \in \mathcal{B}$, and hence, $L((l, x), (k, E)) > 0$. □

Assumption 3. There exists a small set $(\bar{A}, \bar{E}) \subset \mathcal{S} \times \mathcal{B}$ such that $L((l, x), (\bar{A}, \bar{E})) = 1$ for all $(l, x) \in \mathcal{S} \times \mathcal{B}$.

Corollary 8. *The chain W is positive Harris recurrent under Assumption* 3.

Proof. W is a positive chain since it is ψ-irreducible and admits an invariant probability measure $\bar{\mu}$. Moreover, Assumption 3 along with ψ-irreduciblity of W implies that the chain is Harris recurrent (Proposition 9.1.7 in [20]). □

Corollary 9 follows from Theorem 17.0.1 in [20] and proves that the SLLN holds true for chain W under Assumption 3.

Corollary 9. *For any function g defined on $\mathcal{S} \times \mathcal{B}$,*

$$\frac{1}{n} \sum_{j=1}^{n} g(W_j) \to \bar{\mu}(g) \quad \textit{almost surely as} \quad n \to +\infty,$$

if g satisfies $\bar{\mu}(|g|) < +\infty$.

To define the optimal assignment policy, we first introduce some notation. Define $F : \mathcal{B} \to [0, 1]$ as $F(a) := \sum_{j=1}^{r} \mu(j) F_j(a)$, and note that F is a distribution function on $\mathcal{B}$ with $F^{(-1)}(1) := +\infty$ and $F^{(-1)}(0) := -\infty$. Let r^* be defined as in (4.1), with F being the distribution function introduced above. As defined in Section 4.2, a task X_j is labeled type-i if $X_j \in \mathcal{A}_i := (F^{-1}(\pi_i), F^{-1}(\pi_{i-1})]$. Moreover, for a fixed n, let $t_{r_i}^{(n)}$ denote the number of tasks that must arrive until r_i tasks of type i are obtained and define $t^{(n)} := \min\left\{t_{r_1}^{(n)}, t_{r_2}^{(n)}, \cdots, t_{r_k}^{(n)}\right\}$. Note that unlike $U_{r_i}^{(n)}$, $t_{r_i}^{(n)}$ is not distributed negative Binomial, since the task values are no longer assumed to be IID. Unless otherwise mentioned, Assumption 3 holds throughout Section 4.3. Propostion 4 discusses a useful property of $t^{(n)}$.

Proposition 4.

$$\frac{t^{(n)}}{n} \to 1 \quad \textit{almost surely} \quad \textit{as} \quad n \to +\infty \tag{4.13}$$

Proof. It suffices to prove that for all $i \in \{1, 2, \cdots, k\}$, $\frac{t_{r_i}^{(n)}}{n} \to 1$ almost surely as $n \to +\infty$. To this end, arbitrarily fix $i \in \{1, 2, \cdots, k\}$, and note that $\frac{r_i}{n} \to \pi_{i-1} - \pi_i$ as $n \to +\infty$, and hence, it is enough to prove that

$$\frac{t_{r_i}^{(n)}}{r_i} \to \frac{1}{\pi_{i-1} - \pi_i} \quad \text{as} \quad n \to +\infty, \tag{4.14}$$

almost surely. Define $\mathcal{B}_i := \{(Z_j, X_j) : X_j \in \mathcal{A}_i\}$, and note that $t_{r_i}^{(n)}$ is the random time of the r_i^{th} visit to $\mathcal{B}_i$ for the chain W. On the other hand, since

$$\bar{\mu}(I_{\mathcal{B}_i}) = \sum_{u=1}^{r} \mu(u) \left(F_u(F^{-1}(\pi_{i-1})) - F_u(F^{-1}(\pi_i))\right) = \pi_{i-1} - \pi_i,$$

it follows from Corollary 9 that

$$\frac{1}{n} \sum_{j=1}^{n} I_{\mathcal{B}_i}(W_j) \to \pi_{i-1} - \pi_i \quad \textit{as} \quad n \to +\infty,$$

almost surely. Moreover, $t_{r_i}^{(n)} \geq r_i$, and hence, $t_{r_i}^{(n)} \to +\infty$ almost surely as $n + \infty$. Therefore,

$$\frac{1}{t_{r_i}^{(n)}} \sum_{j=1}^{t_{r_i}^{(n)}} I_{\mathcal{B}_i}(W_j) \to \pi_{i-1} - \pi_i \quad \textit{as} \quad n \to +\infty,$$

almost surely, but $\sum_{j=1}^{t_{r_i}^{(n)}} I_{\mathcal{B}_i}(W_j) = r_i$, implying that

$$\frac{r_i}{t_{r_i}^{(n)}} \to \pi_{i-1} - \pi_i \quad \textit{as} \quad n \to +\infty,$$

almost surely, which completes the proof. □

Applying the result in Proposition 4, Theorem 10 presents the optimal policy for $\tau \in [-\infty, r^*)$.

Theorem 10. *Assume that $\tau < r^*$. A policy $\tilde{\phi}_L$ that assigns the j^{th} task to a type-i worker if $X_j \in \mathcal{A}_i$ and $j \leq t^{(n)}$, achieves the infimum in* (4.2).

Proof. The proof is along the same lines as that of Theorem 8 and follows from Corollary 9 and Proposition 4. □

As in the observable distributions case, Theorem 10 along with the results in [8], imply that the policy $\tilde{\phi}_L$ achieves the maximum long-run expected reward per task, while minimizing the risk of the long-run reward per task failing to achieve a given target level. Lemma 15 presents a result, which helps with solving the problem for target values greater than or equal to r^*.

Lemma 15. *Let $X_{(j)}$ denote the j^{th} order statistic of tasks $X_1, X_2, \cdots, X_n$, coming from r different distributions $\{F_1, F_2, \cdots, F_r\}$, where the successive distributions are unobservable and governed by an irreducible*

ergodic Markov chain with invariant distribution μ. It follows that

$$\frac{1}{n}\sum_{j=[n\pi]+1}^{n} X_{(j)} \to \int_{F^{-1}(\pi)}^{+\infty} xF(dx) \quad as \quad n \to +\infty,$$

almost surely for any $\pi \in (0,1)$, where $F(a) := \sum_{j=1}^{r} \mu(j)F_j(a)$.

Proof. Observe that

$$\frac{1}{n}\sum_{i=1}^{n} X_i I_{\{X_i \geq F^{-1}(\pi)\}} \to \bar{\mu}\left(XI_{\{X \geq F^{-1}(\pi)\}}\right) \quad \text{as} \quad n \to +\infty,$$

almost surely by Corollary 9, where

$$\bar{\mu}\left(XI_{\{X \geq F^{-1}(\pi)\}}\right) = \sum_{u=1}^{r} \mu(u) \int_{F^{-1}(\pi)}^{+\infty} xF_u(dx) = \int_{F^{-1}(\pi)}^{+\infty} xF(dx),$$

and hence, it suffices to prove that

$$\frac{1}{n}\Big| \sum_{j=[n\pi]+1}^{n} X_{(j)} - \sum_{i=1}^{n} X_i I_{\{X_i \geq F^{-1}(\pi)\}} \Big| \to 0 \quad \text{as} \quad n \to +\infty, \tag{4.15}$$

almost surely. Define $N(n) := \sum_{i=1}^{n} I_{\{X_i \geq F^{-1}(\pi)\}}$, and note that

$$\frac{1}{n}N(n) \to \bar{\mu}\left(I_{\{X \geq F^{-1}(\pi)\}}\right) = 1 - \pi \quad \text{as} \quad n \to +\infty,$$

almost surely by Corollary 9. Considering the two possible cases (as in Theorem 9), we obtain

$$\frac{1}{n}\Big| \sum_{j=[n\pi]+1}^{n} X_{(j)} - \sum_{i=1}^{n} X_i I_{\{X_i \geq F^{-1}(\pi)\}} \Big| \leq \Big| \frac{n - [n\pi] - N(n)}{n} \Big| \left(Y \vee F^{-1}(\pi)\right) \to 0 \quad \text{as} \quad n \to +\infty,$$

almost surely. □

Using the result in Lemma 15, Theorem 11 solves (4.2) and addresses the problem for target values greater than or equal to r^*.

Theorem 11. *If $\tau \geq r^*$, then*

$$\inf_{\phi \in \Phi} P\left\{ \limsup_{n \to +\infty} \frac{1}{n} R_n^{\phi} \leq \tau \right\} = 1, \tag{4.16}$$

under Assumption 2.

Proof. In light of Corollary 9, Proposition 4, and Lemma 15, the proof is analogous to that of Theorem 9 and Corollary 7. □

According to Theorem 11 and similar to the observable distributions case, the decision maker is indifferent between any two arbitrary admissble policies, when minizing the threshold probability with $\tau \in [r^*, +\infty)$. A prudent choice is to apply the policy $\tilde{\phi_L}$ in this case.

Note that the optimal policy for LTSSAP (i.e., ϕ_L for IID task values and $\tilde{\phi}_L$ for tasks with unobservable distributions) assigns the best α_1 percent of the tasks to the best workers, the second best α_2 percent of the tasks to the second best workers, and so on. Moreover, [8] proves that the same policy maximizes the long-run expected reward per task. A natural question to ask is whether there exist other classes of objective functions, for which this policy is optimal. To answer this question, theorem 12 analyzes SSAP under the following objective function

$$\inf_{\phi\in\Phi} \limsup_{n\to+\infty} P\left\{\frac{1}{n}R_n^{\phi} \leq \tau\right\}, \tag{4.17}$$

and verifies that ϕ_L and $\tilde{\phi}_L$ optimize (4.17) for $\tau \in (-\infty, r^*)$.

Theorem 12. *If $\tau < r^*$, then ϕ_L and $\tilde{\phi}_L$ achieve the optimality in* (4.17) *for the observable and the unobservable distributions case, respectively.*

Proof. Recall from Theorem 8 and Theorem 10 that the reward per task under ϕ_L and $\tilde{\phi}_L$ converges to r^* almost surely as $n \to +\infty$, which implies convergence in probability. Therefore,

$$P\left\{\frac{1}{n}R_n^{\phi_L} \leq \tau\right\} \leq P\left\{\left|\frac{1}{n}R_n^{\phi_L} - r^*\right| > r^* - \tau\right\} \to 0 \quad \text{as} \quad n \to +\infty,$$

implying $\lim_{n\to+\infty} P\left\{\frac{1}{n}R_n^{\phi_L} \leq \tau\right\} = 0$. Therefore, ϕ_L achieves the infimum in (4.17). A similar argument proves the optimality of $\tilde{\phi}_L$ for the unobservable distributions case. □

Theorem 13 proves that if the given target value is greater than r^*, then the decision maker has no preference among the set of admissible policies, under the objective function introduced in (4.17). Recall that this is also the case when solving the LTSSAP.

Theorem 13. *If $\tau > r^*$, then*

$$\inf_{\phi\in\Phi} \limsup_{n\to+\infty} P\left\{\frac{1}{n}R_n^{\phi} \leq \tau\right\} = 1, \tag{4.18}$$

under Assumption 2*, for both the observable and the unobservable distributions case.*

Proof. Recall from Theorem 9 that the long-run reward per task under ϕ_B equals r^* almost surely. On the other hand, no policy can do better than the (infeasible) policy ϕ_B. Therefore,

$$\limsup_{n\to+\infty} \frac{1}{n}R_n^{\phi} \leq \lim_{n\to+\infty} \frac{1}{n}R_n^{\phi_B} = r^* < \tau \tag{4.19}$$

almost surely for any arbitrarily fixed policy $\phi \in \Phi$. It follows from (4.19) and the definition of limsup that

$$P\left\{\liminf_{n\to+\infty}\left\{\frac{1}{n}R_n^{\phi} \leq \tau\right\}\right\} = 1,$$

which implies that

$$1 = \limsup_{n\to+\infty} P\left\{\frac{1}{n}R_n^{\phi} \leq \tau\right\} \geq P\left\{\liminf_{n\to+\infty}\left\{\frac{1}{n}R_n^{\phi} \leq \tau\right\}\right\},$$

for all $\phi \in \Phi$, and hence, (4.18) follows. The same argument proves the result for the unobservable distributions case. □

4.4 Conclusion

Chapter 4 studies the stochastic sequential assignment problem with k fixed worker classes as the number of tasks (denoted by n) approaches infinity, where the goal is to minimize the threshold probability. Two versions of the problem are studied based on the distribution function of the task values being observable or unobservable. Simple stationary optimal policies with $k-1$ fixed breakpoints are presented for both problems, as opposed to the policy described in [12], where the number of breakpoints increases with n, and the breakpoints are recalculated upon the arrival of each task. Apart from being a simple replacement for the optimal policy defined in [12], this optimal policy not only maximizes the long-run expected reward per task but also minimizes the risk of failing to achieve a given threshold value. Therefore, it diverges from the risk-neutral objective function studied in the existing SSAP literature.

This work also incorporates a Markov-modulated dependency between task values and uncertainty in the task distribution functions simultaneously into the model. Further research is required to address other types of uncertainty in the SSAP with a large number of tasks. One example can be the case where the number of arriving tasks is not known apriori and follows a given probability distribution. Another challenge is shifting the attention from the IID sequence of tasks to a more general case with dependent task values, where the value of the current task depends on the whole sequence of its preceding tasks.

Chapter 5

Stochastic Sequential Assignment Problem with Random Success Rates

5.1 Introduction

Consider the sequential stochastic assignment problem (SSAP) introduced in [12], where n workers are available to perform n IID sequentially-arriving tasks. The random variable X_j denotes the j^{th} task value, and a value (success rate) p_i is associated with each worker. Whenever the i^{th} worker is assigned to the j^{th} task, that worker becomes unavailable for future assignments, with $p_i x_j$ denoting the expected reward due to this assignment. The objective is to assign these n workers to n tasks so as to maximize the expected total reward. It is shown in [12] that there exists numbers

$$-\infty = a_{0,n} \leq a_{1,n} \leq a_{2,n} \leq \cdots \leq a_{n,n} = +\infty,$$

such that the optimal choice in the initial stage is to assign the i^{th} best available worker (i.e., the one with the i^{th} highest success rate) if the random variable X_1 falls within the i^{th} highest interval. The optimal expected total reward obtained from assigning all the n tasks to workers is $\sum_{i=1}^{n} p_i a_{i,n+1}$, and hence, $a_{i,n+1}$ is the expected value of the quantity assigned to the i^{th} best worker. Moreover, the $a_{i,n}$ are independent of the worker values and depend on the task values' distribution function. Specifically, these breakpoints are computed recursively from

$$a_{i,n+1} = \int_{a_{i-1,n}}^{a_{i,n}} u\, dG(u) + a_{i-1,n} G(a_{i-1,n}) + a_{i,n} \left(1 - G(a_{i,n})\right), \tag{5.1}$$

for $i = 1, 2, \cdots, n$, with $a_{0,n} = -\infty$ and $a_{n,n} = +\infty$, where G is the cumulative distribution function of task values [12].

The SSAP has applications in several areas, and various extensions to the problem have been discussed in the literature. For example, [23] studies a variation of the SSAP in aviation security screening systems where sequentially-arriving passengers are assigned to the available screening devices, while [31] addresses the problem of allocating sequentially-arriving kidneys to patients on a waiting list. Another application of the SSAP is the asset selling problem [5], where one needs to choose the best offers out of a sequence of bids from potential buyers. Moreover, [2] analyzes a problem where IID tasks arrive sequentially at random times

according to a renewal process and must be assigned to a fixed set of workers. The same problem is studied by [3], where various arrival-time distributions are considered. [28] also examines SSAP where tasks arrive according to a Poisson process and are assigned random deadlines. [22] considers a variation of SSAP in which the number of tasks is unknown until after the final arrival and follows a given probability distribution. Asymptotic behavior of a special case of SSAP, where worker values are only allowed to take on values of zero or one, is analyzed in [7], and an optimal policy has been established as the number of arriving tasks approaches infinity. A generalization of this problem is studied by [8], where workers are allowed to take on any values. Also, [8] considers this model under the assumption of incomplete information, where task distributions are unobservable to the decision-maker, and presents a stationary optimal policy to achieve the long-run expected reward per task. [9] investigates the problem under a different objective function, called *threshold criterion*, which minimizes the probability of the total reward failing to achieve a specified (target) value.

The existing SSAP literature assumes the worker values to be deterministic and known in advance. This dissertation chapter studies an extension of SSAP, in which the success rates are random variables, taking on new values upon each task arrival. For example, consider the asset selling problem, where a sequence of bids (tasks) arrive from potential buyers, and a set of items (workers) are available to be sold to these buyers so as to maximize the expected total profit. A time-dependent random price is associated with each item. Each item price is dependent on the economic conditions upon the arrival of the offer and hence takes on a new value upon each task arrival, since economic conditions vary from time to time. This can be modelled as a set of distribution functions, governed by a Markov chain, which generate item prices. Markov chain transitions represent the variations in economic conditions (equivalently, switching from one price distribution to another). Specifically, let Z_j denote the state of the Markov chain with state space $\mathcal{S} = \{1, 2, \cdots, r\}$ at time period j, then $Z_j = k$ implies that item prices upon the arrival of offer j are derived from distribution function F_k, which corresponds to the economic conditions at time j.

The remainder of this chapter is organized as follows. Section 5.2 presents mathematical models of four classes of SSAP's with random success rates, providing closed form expression for optimal assignment policies so as to maximize the expected total reward. Section 5.3 offers concluding comments and future research directions.

5.2 Model Description

Consider the original SSAP introduced by [12], where n workers are available to perform n IID sequentially-arriving tasks so as to maximize the expected total reward. A random variable X_j denotes the value of the j^{th} task that arrives during time period j, with a fixed value (or success rate) p_i associated with worker

i. If the i^{th} worker is assigned to the j^{th} task with observed value x_j, the worker becomes unavailable for future assignments, and the expected reward due to this assignment is given by $p_i x_j$. In this dissertation, we study various versions of the SSAP, where worker success rates are no longer assumed to be fixed or deterministic. Specifically, four models are introduced which analyze this problem from different aspects; e.g., the distribution function of worker values being identical or distinct, the worker values in a given time period being dependent on those in the preceding time periods (or within the same time period), worker values being deterministic but assumed to be functions of time (possibly deteriorating with time), and task values being independent or dependent on each other. Worker values are observable to the decision-maker (i.e., the person who makes the assignments) and take on new values upon each task arrival. Section 5.2.1 examines two models with time-dependent IID worker values.

5.2.1 Model I

Assume that n workers are available to be assigned to n tasks that arrive sequentially, with X_j denoting the j^{th} task value. Let $P_k = (p_k(1), p_k(2), \cdots, p_k(n-k+1))$ be the vector of success rates at time period k, with $p_k(i)$ being the success rate of the i^{th} worker at that time. Two versions of Model I are discussed. The first version, labeled Model I.1, considers the task values to be IID. For an arbitrarily fixed time period k, $\{p_k(i)\}_{i=1}^{n-k+1}$ is assumed to be an IID sequence of random variables with *pdf* f, where the success rates are also independent of task values. Moreover, the vector of success rates at time period k is independent of $\{P_i\}_{i=1}^{k-1}$. In other words, worker values are no longer considered to be fixed and known in advance. In fact, success rates in Model I are time-dependent and random variables. Specifically, a new and distinct vector of worker values (with IID elements) is revealed to the decision-maker upon the arrival of each task, where one of the available workers is chosen to perform that task. The objective is to maximize the expected total reward obtained from all the n assignments. Theorem 14 (also mentioned in [18]) discusses the optimal assignment policy for Model I.1 along with the optimal expected total reward.

Theorem 14. *The optimal decision at each time period in Model I.1 is to assign the best available worker (i.e., the one with the highest success rate) to the current task. The resulting optimal expected total reward is then given by* $E_{\phi^*}[R_n] = E[X] \sum_{k=1}^{n} E[M_p(k)]$, *where* $M_p(k) := \max_{i=1,\cdots,n-k+1} p_k(i)$, R_n *is the total reward over the last* n *time periods, and* ϕ^* *is the optimal assignment policy.*

Proof. The proof is by induction on n, starting from $n = 2$. The optimal conditional expected total reward

for the base case $n = 2$ upon observing the first task value is given by

$$\begin{aligned} E_{\phi^*}[R_2|X_1] &= \max\{p_1(1)X_1 + E[X]E[p_2(2)],\, p_1(2)X_1 + E[X]E[p_2(1)]\} \\ &= \max\{p_1(1)X_1 + E[X]E[M_p(2)],\, p_1(2)X_1 + E[X]E[M_p(2)]\} \\ &= M_p(1)X_1 + E[X]E[M_p(2)], \end{aligned} \tag{5.2}$$

where the second equality follows since worker values are considered to be IID at a given time period; specifically, $E[p_2(1)] = E[p_2(2)]$ since $p_2(1)$ and $p_2(2)$ are identically distributed, and they both equal $E[M_p(2)]$ for $n = 2$, by definition. Observe that (5.2) implies that the worker with the highest success rate is assigned to the first task. Hence, $E_{\phi^*}[R_2] = E[X]\sum_{k=1}^{2} E[M_p(k)]$, and the statement is proven for $n = 2$. To proceed, assume that the statement holds true for some $n \geq 2$, with $E_{\phi^*}[R_n|X_1] = M_p(1)X_1 + E[X]\sum_{k=2}^{n} E[M_p(k)]$, where ϕ^* is the optimal assignment policy for the last n time periods. Upon observing the first task value, the optimal conditional expected total reward for the case with $n+1$ workers is given by

$$\begin{aligned} E_{\phi^*}[R_{n+1}|X_1] &= \max_{i=1,\cdots,n+1}\{p_1(i)X_1 + E_{\phi^*}[R_n]\} \\ &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1 + E[X]\sum_{k=2}^{n+1} E[M_p(k)]\right\} \\ &= M_p(1)X_1 + E[X]\sum_{k=2}^{n+1} E[M_p(k)], \end{aligned} \tag{5.3}$$

where the first equality follows since success rates are IID at each time period so that the expected total reward over the last n time periods is independent of the index of the worker chosen at time period 1. The second equality follows from the induction assumption. Due to (5.3), the optimal decision upon the arrival of X_1 is to choose the best available worker, and $E_{\phi^*}[R_{n+1}] = E[X]\sum_{k=1}^{n+1} E[M_p(k)]$. □

Note that f, the *pdf* of success rates, is considered to be time-independent so far. Corollary 10 studies Model I.1, while relaxing this assumption.

Corollary 10. *In Model I.1, suppose that for an arbitrarily fixed time period k, $p_k(i) \sim f_k$ for $i = 1, 2, \cdots, n-k+1$. The optimal decision at each time period is to assign the best available worker, and the optimal expected total reward is given by $E[X]\sum_{k=1}^{n} E[M_p(k)]$.*

Proof. The proof is by induction on n and is omitted due to analogy to that of Theorem 14. □

The second model, labeled Model I.2, generalizes Model I.1 to the case where task values are dependent on one another. Assume that the sequence of random variables $\{X_j,\, j = 1, 2, \cdots, n\}$ is defined on the probability space $(\Omega, \mathcal{F}, \mathcal{P})$ adapted to a filteration $\{\mathcal{F}_j,\, j = 1, 2, \cdots, n\}$ of $\mathcal{F}$. Other assumptions of Model

I.1 still hold for Model I.2. Specifically, for an arbitrarily fixed time period k, $\{p_k(i)\}_{i=1}^{n-k+1}$ is an IID sequence of random variables with *pdf* f, with the success rates being independent of task values. Moreover, the vector of success rates at time period k is independent of $\{P_i\}_{i=1}^{k-1}$. Theorem 15 presents the optimal assignment policy to maximize the expected total reward for Model I.2.

Theorem 15. *The optimal decision at each time period in Model I.2 is to assign the best available worker to the current task. The resulting optimal expected total reward is then given by* $E_{\phi^*}[R_n] = \sum_{k=1}^{n} E[X_k]E[M_p(k)]$.

Proof. The proof is by induction on n, starting from $n = 2$. The optimal conditional expected total reward for the base case $n = 2$ is given by

$$\begin{aligned} E_{\phi^*}[R_2|X_1, P_1] &= \max\{E[p_1(1)X_1 + p_2(2)X_2|X_1, P_1], E[p_1(2)X_1 + p_2(1)X_2|X_1, P_1]\} \\ &= \max\{p_1(1)X_1 + E[p_2(2)X_2|\mathcal{F}_1], p_1(2)X_1 + E[p_2(1)X_2|\mathcal{F}_1]\} \\ &= \max\{p_1(1)X_1 + E[M_p(2)]E[X_2|\mathcal{F}_1], p_1(2)X_1 + E[M_p(2)]E[X_2|\mathcal{F}_1]\} \\ &= M_p(1)X_1 + E[M_p(2)]E[X_2|\mathcal{F}_1], \end{aligned} \tag{5.4}$$

where the second equality follows since $\sigma(P_2X_2, X_1)$ is independent of $\sigma(P_1)$, with $\sigma(Y)$ denoting the σ-algebra generated by Y. The third equality follows since $\sigma(P_2)$ is independent of $\sigma(X_1, X_2) = \mathcal{F}_2$. Hence, $E_{\phi^*}[R_2] = \sum_{k=1}^{2} E[X_k]E[M_p(k)]$, and the statement is proven for $n = 2$. Now suppose that the statement of Theorem 15 holds true for some $n \geq 2$, where the optimal assignment policy over the last n stages is denoted by ϕ^* and $E_{\phi^*}[R_n|X_1, P_1] = M_p(1)X_1 + \sum_{k=2}^{n} E[M_p(k)]E[X_k|\mathcal{F}_1]$. To prove the claim, note that the optimal conditional expected total reward for the case with $n + 1$ workers is given by

$$\begin{aligned} E_{\phi^*}[R_{n+1}|X_1, P_1] &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_1, P_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[R_{n+1\setminus\{i\}}|\mathcal{F}_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[E_{\phi^*}\left[R_{n+1\setminus\{i\}}|\mathcal{F}_2\right]|\mathcal{F}_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[M_p(2)X_2 + \sum_{k=3}^{n+1} E[M_p(k)]E[X_k|\mathcal{F}_2]\Big|\mathcal{F}_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E[M_p(2)]E[X_2|\mathcal{F}_1] + \sum_{k=3}^{n+1} E[M_p(k)]E[X_k|\mathcal{F}_1]\right\} \\ &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + \sum_{k=2}^{n+1} E[M_p(k)]E[X_k|\mathcal{F}_1]\right\} \\ &= M_p(1)X_1 + \sum_{k=2}^{n+1} E[M_p(k)]E[X_k|\mathcal{F}_1], \end{aligned} \tag{5.5}$$

where $R_{n\setminus\{i\}}$ denotes the total reward over the last n time periods (specifically, time periods $2, 3, \cdots, n+1$ in this case) after removing the i^{th} worker. The second equality follows since $\sigma(R_{n\setminus\{i\}}, X_1)$ is independent of $\sigma(P_1)$, and the fourth equality is a direct result of the induction assumption. The fifth equality follows due to the fact that $\sigma(M_p(2))$ is independent of $\sigma(X_1, X_2) = \mathcal{F}_2$. Note that $\sum_{k=2}^{n+1} E[M_p(k)]E[X_k|\mathcal{F}_1]$ (i.e., the second term in the equality before the last in (5.5)) is independent of i; therefore, it can be deduced from (5.5) that the optimal decision upon the arrival of the first task is to choose the best available worker, and $E_{\phi^*}[R_{n+1}] = \sum_{k=1}^{n+1} E[M_p(k)]\, E[X_k]$. □

5.2.2 Model II

Incorporating the randomness of worker values into the problem, Section 5.2.2 discusses three distinct models while modifying the assumption that the worker values are independent at a given time period or within different time periods. Model II.1 studies a variation of Model I.1, where the independence assumption between vector of success rates at different time periods is relaxed. Specifically, the sequence of arriving tasks is considered to be IID, while the success rate of worker i through time, $\{p_k(i)\}_{k=1}^{n}$, forms a sequence of dependent random variables, for any arbitrarily fixed $i = 1, 2, \cdots, n$. Moreover, the set of worker success rates are assumed to be exchangeable (i.e., identically distributed bu not necessarily IID), given the past values. Theorem 16 presents the optimal assignment policy to maximize the expected total reward for this model.

Theorem 16. *The optimal decision at each time period in Model II.1 is to assign the best available worker to the current task. The resulting optimal expected total reward is then given by* $E_{\phi^*}[R_n] = E[X] \sum_{k=1}^{n} E[M_p(k)]$.

Proof. The proof is by induction on n. Starting from $n = 2$, the optimal conditional expected total reward after observing the first task and worker values is given by

$$\begin{aligned} E_{\phi^*}[R_2|X_1, P_1] &= \max\{E[p_1(1)X_1 + p_2(2)X_2|X_1, P_1],\, E[p_1(2)X_1 + p_2(1)X_2|X_1, P_1]\} \\ &= \max\{p_1(1)X_1 + E[p_2(2)X_2|P_1],\, p_1(2)X_1 + E[p_2(1)X_2|P_1]\} \\ &= \max\{p_1(1)X_1 + E[p_2(2)|P_1]E[X],\, p_1(2)X_1 + E[p_2(1)|P_1]E[X]\} \\ &= \max_{i=1,2}\{p_1(i)X_1 + E[M_p(2)|P_1]E[X]\} \\ &= M_p(1)X_1 + E[M_p(2)|P_1]E[X], \end{aligned} \tag{5.6}$$

where the second equality follows since $\sigma(P_2X_2, P_1)$ is independent of $\sigma(X_1)$, and the third equality follows since $\sigma(X_2)$ is independent of $\sigma(P_1, P_2)$. It can be deduced from (5.6) that the optimal decision upon the arrival of the first task is to assign it to the best available worker, since $E[M_p(2)|P_1]E[X]$ is independent of

i. Now suppose that the statement of Theorem 16 holds true for some $n \geq 2$, where the optimal conditional expected total reward is given by $E_{\phi^*}[R_n|X_1, P_1] = M_p(1)X_1 + E[X]\sum_{k=2}^{n} E[M_p(k)|P_1]$, with ϕ^* denoting the optimal assignment policy over the last n stages. To prove the claim, note that the optimal conditional expected total reward for the case with $n+1$ workers, after observing the first task and worker values, is given by

$$\begin{aligned}
E_{\phi^*}[R_{n+1}|X_1, P_1] &= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_1, P_1\right]\right\} \\
&= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_2, P_2\right]|X_1, P_1\right]\right\} \\
&= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E_{\phi^*}\left[M_p(2)X_2 + E[X]\sum_{k=3}^{n+1} E[M_p(k)|P_2]\Big|X_1, P_1\right]\right\} \\
&= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E[X]E[M_p(2)|P_1] + E[X]\sum_{k=3}^{n+1} E[M_p(k)|P_1]\right\} \\
&= \max_{i=1,\cdots,n+1} \left\{p_1(i)X_1 + E[X]\sum_{k=2}^{n+1} E[M_p(k)|P_1]\right\} \\
&= M_p(1)X_1 + E[X]\sum_{k=2}^{n+1} E[M_p(k)|P_1],
\end{aligned} \tag{5.7}$$

where the third equality follows from the induction assumption. It follows from (5.7) that it is optimal to assign the best available worker to the first arriving task and that $E_{\phi^*}[R_{n+1}] = E[X]\sum_{k=1}^{n+1} E[M_p(k)]$. □

Model II.2 analyzes the case where IID sequentially-arriving tasks are assigned to a set of workers, with $\{p_k(i)\}_{k=1}^{n}$ forming a Martingale for arbitrarily fixed $i = 1, 2, \cdots, n$. Worker success rates at a given time period are no longer assumed to be IID. In fact, any kind of dependency is allowed. However, it is assumed that the worker values (although being random variables) have a fixed order throughout the problem. Specifically, suppose that the order of success rates observed in P_1 is assumed to be fixed throughout the problem. For simplicity, once this order is revealed at the beginning of assignments, the workers are re-labeled such that $p_1(i)$ would correspond to the worker with the i^{th} smallest value. Theorem 17 explains how to tackle this problem in order to obtain the optimal expected total reward.

Theorem 17. *The optimal decision in Model II.2 is to assign the i^{th} best worker to the first task with value X_1 if it falls within the i^{th} highest interval formed by the $\{a_{i,n}\}_{i=1}^{n-1}$, as calculated in* (5.1)*. The resulting optimal expected total reward is then given by $E_{\phi^*}[R_n] = \sum_{i=1}^{n} a_{i,n+1}E[p_1(i)]$.*

Proof. The proof is by induction on n. Starting from $n = 2$, the optimal conditional expected total reward

after observing the first task and worker values is given by

$$\begin{aligned} E_{\phi^*}\left[R_2|X_1,P_1\right] &= \max\left\{E\left[p_1(1)X_1+p_2(2)X_2|X_1,P_1\right],\, E\left[p_1(2)X_1+p_2(1)X_2|X_1,P_1\right]\right\} \\ &= \max\left\{p_1(1)X_1+E[p_2(2)X_2|P_1],\, p_1(2)X_1+E[p_2(1)X_2|P_1]\right\} \\ &= \max\left\{p_1(1)X_1+E[p_2(2)|P_1]E[X],\, p_1(2)X_1+E[p_2(1)|P_1]E[X]\right\} \\ &= \max\left\{p_1(1)X_1+p_1(2)E[X],\, p_1(2)X_1+p_1(1)E[X]\right\} \\ &= p_1(1)\left(X_1\wedge E[X]\right)+p_1(2)\left(X_1\vee E[X]\right), \end{aligned} \tag{5.8}$$

where the second equality follows since $\sigma(P_2X_2,P_1)$ is independent of $\sigma(X_1)$. The third equality follows since $\sigma(X_2)$ is independent of $\sigma(P_1,P_2)$, while the fourth equality is a direct result of the Martingale property. Let $\vee$ and $\wedge$ denote the maximum and the minimum, respectively. Note that according to (5.8), the optimal action to take upon the arrival of the first task is to compare its value with $a_{1,2}:=E[X]$ and assign it to the worker with value $p_1(1)$ if $X_1=X_1\wedge E[X]$ and to $p_1(2)$ otherwise. As a consequence of (5.8), it follows that

$$\begin{aligned} E_{\phi^*}\left[R_2|P_1\right] &= p_1(1)E\left[X_1\wedge E[X]\right]+p_1(2)E\left[X_1\vee E[X]\right] \\ &= p_1(1)a_{1,3}+p_1(2)a_{2,3}. \end{aligned}$$

Now suppose that the statement of Theorem 17 holds true for some $n\geq 2$, where the optimal assignment policy over the last n stages is denoted by ϕ^* and $E_{\phi^*}\left[R_n|P_1\right]=\sum_{j=1}^{n}a_{i,n+1}p_1(j)$. To prove the claim, note that the optimal conditional expected total reward, upon the arrival of first task, for the case with $n+1$ workers is given by

$$\begin{aligned} E_{\phi^*}\left[R_{n+1}|X_1,P_1\right] &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_1,P_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[R_{n+1\setminus\{i\}}|P_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[E_{\phi^*}\left[R_{n+1\setminus\{i\}}|P_2\right]|P_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[\sum_{j=1}^{i-1}a_{j,n+1}p_2(j)+\sum_{k=i}^{n}a_{k,n+1}p_2(k+1)\Big|P_1\right]\right\} \\ &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+\sum_{j=1}^{i-1}a_{j,n+1}p_1(j)+\sum_{k=i}^{n}a_{k,n+1}p_1(k+1)\right\}, \end{aligned} \tag{5.9}$$

where the fourth equality is a result of the induction assumption and the assumption that the order of worker success rates is fixed (i.e., $p_2(l)$ is an increasing function of l). The last equality follows from the Martingale property. Recall that $\{p_1(l)\}_{l=1}^{n+1}$ is an increasing sequence of size $n+1$, and $\{a_{m,n+1}\}_{m=1}^{n}$ is an increasing

sequence of size n. According to Hardy's theorem, the optimal decision upon the arrival of the first task with value X_1 is to match X_1 with the i^{th} best worker if it falls within the i^{th} highest interval formed by $\{a_{m,n+1}\}_{m=1}^{n}$. Therefore,

$$E_{\phi^*}\left[R_{n+1}|P_1\right] = \sum_{i=1}^{n+1} a_{i,n+2} p_1(i),$$

since $\{a_{i,n+2}\}_{i=1}^{n+1}$ is the set of ordered values of X_1 and $\{a_{m,n+1}\}_{m=1}^{n}$ (see (5.1)).

□

In a similar fashion, Model II.2 and the result of Theorem 17 can be extended to the case of incomplete information, where the actual worker values are unobservable to the decision-maker; however, the decision-maker knows the order of these values, which is assumed to be fixed throughout the problem. Specifically, the optimal assignment policy and the optimal expected total reward in this case is the same as that presented in Theorem 17.

Model II.3 generalizes Model II.2 to the case where the sequence of arriving tasks is no longer IID, while keeping all the other assumptions of Model II.2. To obtain the optimal policy for this model, a set of breakpoints must be calculated upon the arrival of each task, and the value of that task is compared with these breakpoints to determine the index of the worker to assign to the task. These breakpoints are computed in a recursive manner, as the breakpoints $\{a_{i,n}\}$ computed in (5.1) that characterize the optimal policy of Theorem 17. However, the set of breakpoints upon the arrival of task j (for any arbitrarily fixed j) are random variables, the values of which are realized upon observing the value of X_j. These breakpoints, that make up the optimal assignment policy for Model II.3, are computed as follows. Recall that n indicates the total number of tasks to arrive. For any $r \in \{1, 2, \cdots, n+1\}$, define the random variable $A_{m,r}^{(n)}$ as

$$A_{m,r}^{(n)} = \begin{cases} -\infty & m = 0 \\ E\left[\tilde{A}_{m,r}^{(n)} \middle| \mathcal{F}_{r-1}\right] & 1 \leq m \leq n-r+1 \\ +\infty & m > n-r+1 \end{cases}, \tag{5.10}$$

where $\tilde{A}_{m,r}^{(n)} = \left(X_r \vee A_{m-1,r+1}^{(n)}\right) \wedge A_{m,r+1}^{(n)}$. Theorem 18 discusses the optimal assignment policy with a threshold structure for Model II.3 with random success rates, where the threshold values are computed iteratively via (5.10).

Theorem 18. *The optimal decision in Model II.3 at time period r, for $r = 1, 2, \cdots, n$, is to assign the i^{th} best worker to the r^{th} task with value X_r if it falls within the i^{th} highest interval formed by the $\left\{A_{m,r+1}^{(n)}\right\}_{m=1}^{n-r}$, as computed in* (5.10). *The resulting optimal expected total reward is then given by $E_{\phi^*}\left[R_n|X_1, P_1\right] =$*

$\sum_{i=1}^{n} p_1(i)\tilde{A}_{i,1}^{(n)}$.

Proof. The proof is by induction on n. Starting from $n = 2$, the optimal conditional expected total reward after observing the first task and worker values is given by

$$\begin{aligned}
E_{\phi^*}\left[R_2|X_1,P_1\right] &= \max\left\{E\left[p_1(1)X_1+p_2(2)X_2|X_1,P_1\right],\, E\left[p_1(2)X_1+p_2(1)X_2|X_1,P_1\right]\right\}\\
&= \max\left\{p_1(1)X_1+E[p_2(2)X_2|X_1,P_1],\, p_1(2)X_1+E[p_2(1)X_2|X_1,P_1]\right\}\\
&= \max\left\{p_1(1)X_1+E[p_2(2)|P_1]E[X_2|X_1],\, p_1(2)X_1+E[p_2(1)|P_1]E[X_2|X_1]\right\}\\
&= \max\left\{p_1(1)X_1+p_1(2)E[X_2|X_1],\, p_1(2)X_1+p_1(1)E[X_2|X_1]\right\}\\
&= p_1(1)\left(X_1\wedge A_{1,2}^{(2)}\right)+p_1(2)\left(X_1\vee A_{1,2}^{(2)}\right)\\
&= p_1(1)\tilde{A}_{1,1}^{(2)}+p_1(2)\tilde{A}_{2,1}^{(2)},
\end{aligned} \tag{5.11}$$

where the fourth equality is a direct result of the Martingale property. Note that the optimal action to take upon the arrival of the first task with value X_1 is to compare its value with $A_{1,2}^{(2)}$ and assign it to the first worker if $X_1 = X_1 \wedge A_{1,2}^{(2)}$ and to the second worker otherwise. The last equality in (5.11) implies that $\tilde{A}_{m,1}^{(2)}$ is the expected value of the task assigned to the m^{th} smallest worker, upon the arrival (and observing the value) of X_1, when there are a total of $n = 2$ tasks to arrive. Now suppose that the statement of Theorem 17 holds true for some $n \geq 2$ and the optimal assignment policy over the last n stages is denoted by ϕ^*. To prove the claim, note that the optimal conditional expected total reward for the case with $n+1$ workers, after observing the first task and worker values, is given by

$$\begin{aligned}
E_{\phi^*}\left[R_{n+1}|X_1,P_1\right] &= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_1,P_1\right]\right\}\\
&= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[E_{\phi^*}\left[R_{n+1\setminus\{i\}}|X_2,P_2\right]|X_1,P_1\right]\right\}\\
&= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+E_{\phi^*}\left[\sum_{j=1}^{i-1}p_2(j)\tilde{A}_{j,2}^{(n+1)}+\sum_{k=i}^{n}p_2(k+1)\tilde{A}_{k,2}^{(n+1)}\Big|X_1,P_1\right]\right\}\\
&= \max_{i=1,\cdots,n+1}\left\{p_1(i)X_1+\sum_{j=1}^{i-1}p_1(j)A_{j,2}^{(n+1)}+\sum_{k=i}^{n}p_1(k+1)A_{k,2}^{(n+1)}\right\}\\
&= \sum_{i=1}^{n+1}p_1(i)\tilde{A}_{i,1}^{(n+1)},
\end{aligned} \tag{5.12}$$

where the last equality follows since $\left\{\tilde{A}_{m,1}^{(n+1)}\right\}_{m=1}^{n+1}$ is the set of ordered values of

$$\left\{X_1, A_{k,2}^{(n+1)}, \text{ for all } \ k=1,2,\cdots,n\right\},$$

by definition. Note that $\{p_1(l)\}_{l=1}^{n+1}$ is an increasing sequence of size $n+1$, and $\left\{A_{m,2}^{(n+1)}\right\}_{m=1}^{n}$ is an increasing sequence of size n. Therefore, according to Hardy's theorem, the optimal decision upon the arrival of the first task with value X_1 is to match it with the i^{th} best worker if it falls within the i^{th} highest interval formed by $\left\{A_{m,2}^{(n+1)}\right\}_{m=1}^{n}$. The last equality in (5.12) implies that $\tilde{A}_{m,1}^{(n+1)}$ is the expected value of the task assigned to the m^{th} smallest worker, upon the arrival (and observing the value) of X_1, when there are a total of $n+1$ tasks to arrive. □

5.2.3 Model III

This section considers two models, in which worker values vary with time and form two distinct classes based on their values. Assume that there are n IID sequentially arriving tasks (and hence, n remaining time periods), while m identical workers ($m \leq n$) with non-zero values are available to be assigned to these tasks. Model III.1 considers the m worker values to be deterministic and time-dependent. Specifically, if the j^{th} task with value X_j arrives at time t_j, then the non-zero worker values at that time are given by $p(t_j)$. As a result, once a task arrives, the decision-maker must decide whether to accept the task (i.e., assign it to one of the available workers with a non-zero value) or to reject it (i.e., assign it to a worker with value zero). Let $V_j^{(l)}(x)$ be the maximum conditional expected total reward obtained over the last $n-j+1$ time periods (specifically, time periods $j, j+1, \cdots, n$), when there are l workers with non-zero values available upon the arrival of the j^{th} task with value $X_j = x$. Moreover, define $V_j^{(l)} := E\left[V_j^{(l)}(X)\right]$ for any $j = 1, 2, \cdots, n$ and $l = 1, 2, \cdots, n-j+1$. Theorem 19 discusses the optimal assignment policy for Model III.1, which has a threshold structure.

Theorem 19. *There exists thresholds $y^{(m)}(t), y^{(m-1)}(t), \cdots, y^{(1)}(t)$ such that if the j^{th} task arrives at time t_j and l workers are available for assignment at that time, then the optimal decision is to accept the j^{th} task with value X_j if and only if*

$$X_j \geq y^{(l)}(t_j), \tag{5.13}$$

where

$$y^{(l)}(t_j) := \frac{V_{j+1}^{(l)} - V_{j+1}^{(l-1)}}{p(t_j)}.$$

Moreover, for any fixed $l = 1, 2, \cdots, m$, $\left\{V_j^{(l)}\right\}_{j=1}^{n-l+1}$ is computed recursively from $\left\{V_{\tilde{j}}^{(l-1)}\right\}_{\tilde{j}=1}^{n-l+2}$ by

$$V_{n-l+1}^{(l)} = p(t_{n-l+1})E[X] + V_{n-l+2}^{(l-1)}, \tag{5.14}$$

and

$$V_j^{(l)} = p(t_j)E[X \vee y^{(l)}(t_j)] + V_{j+1}^{(l-1)} \quad \textit{for} \quad j = 1, 2, \cdots, n-l, \tag{5.15}$$

with $V_j^{(0)} := 0$ *for all* j.

Proof. To prove (5.14), note that at time period $n-l+1$, l tasks have yet to arrive (specifically, with values $X_{n-l+1}, X_{n-l+2}, \cdots, X_n$) and considering that l workers are available to be assigned to these arrving tasks, the best decision upon the arrival of each of these l tasks is to accept the task and assign it to one of the available workers upon its arrival. Therefore, once the task with value $X_{n-l+1} = x$ arrives, it is assigned to a worker of value $p(t_{n-l+1})$, and $l-1$ tasks (specifically, with values $X_{n-l+2}, X_{n-l+3}, \cdots, X_n$) are remained to be assigned to $l-1$ identical remaining workers. In other words,

$$V_{n-l+1}^{(l)}(x) = p(t_{n-l+1})x + V_{n-l+2}^{(l-1)},$$

and (5.14) follows. Proving (5.15) is along the same lines; however, the number of remaining tasks in this case is not equal to the number of available workers. Therefore, upon the arrival of the j^{th} task with value X_j at time t_j, the decision-maker must decide whether to accept the task or discard it. In mathematical form, this is shown by

$$\begin{aligned} V_j^{(l)}(x) &= \max\left\{p(t_j)x + V_{j+1}^{(l-1)},\, V_{j+1}^{(l)}\right\} \\ &= p(t_j)\left(x \vee y^{(l)}(t_j)\right) + V_{j+1}^{(l-1)}, \end{aligned}$$

which results in (5.13) and (5.15). Specifically, the j^{th} task with value $X_j = x$ is accepted if $p(t_j)x + V_{j+1}^{(l-1)} \geq V_{j+1}^{(l)}$, which is equivalent to (5.13). Also, (5.15) follows since

$$V_j^{(l)} = E\left[V_j^{(l)}(X)\right] = p(t_j)E[X \vee y^{(l)}(t_j)] + V_{j+1}^{(l-1)}.$$

□

Observe that in order to obtain the optimal threshold values for a problem with m available workers in Model III.1, the threshold values for problems with $l = 1, 2, \cdots, m-1$ available workers must be computed in advance. Moreover, the threshold values computed in (5.13) are functions of arrival times, task distributions, number of available workers, and worker values, as opposed to the models studied in the previous sections where the thresholds only depend on arrival times and task distributions.

Under the same setting and assumptions, Model III.2 extends the results of Model III.1 to the case where the non-zero identical worker values are random variables with time-dependent *pdf*'s (i.e., $p(t_j) \sim f_j$). Specifically, the worker values at time period j only become observable upon the arrival of task j with value

X_j. Moreover, task and worker values are no longer assumed to independent. Corollary 11 generalizes the results of Theorem 19 to the case with random worker values.

Corollary 11. *There exists thresholds $\tilde{y}^{(m)}(t), \tilde{y}^{(m-1)}(t), \cdots, \tilde{y}^{(1)}(t)$ such that if the j^{th} task arrives at time t_j and l workers with random values are available for assignment at that time, then the optimal decision is to accept the j^{th} task with value X_j if and only if*

$$W_j \geq \tilde{y}^{(l)}(t_j), \tag{5.16}$$

where $W_j := p(t_j)X_j$ and $\tilde{y}^{(l)}(t_j) := V_{j+1}^{(l)} - V_{j+1}^{(l-1)}$. Moreover, for any fixed $l = 1, 2, \cdots, m$, $\left\{V_j^{(l)}\right\}_{j=1}^{n-l+1}$ is computed recursively from $\left\{V_{\tilde{j}}^{(l-1)}\right\}_{\tilde{j}=1}^{n-l+2}$ by

$$V_{n-l+1}^{(l)} = E[W_{n-l+1}] + V_{n-l+2}^{(l-1)}, \tag{5.17}$$

and

$$V_j^{(l)} = E[W_j \vee \tilde{y}^{(l)}(t_j)] + V_{j+1}^{(l-1)} \quad \textit{for} \quad j = 1, 2, \cdots, n-l, \tag{5.18}$$

with $V_j^{(0)} := 0$ for all j.

Proof. The proof is an extension of that of Theorem 19 and is therefore omitted. □

5.2.4 Model IV

Section 5.2.3 studies the SSAP with random worker values, where the workers form two distinct classes based on their success rates; however, the success rate of one of the classes is assumed to be zero. Section 5.2.4 extends this problem to the case where workers are allowed to form two or more classes with non-zero success rates and worker values at a given time are derived from distinct distribution functions. Specifically, Model IV.1 studies a SSAP with $n = 2$, where two tasks arrive sequentially to be assigned to two workers, with random values α and β. Worker values at time period j are denoted by $\alpha_j \sim f_\alpha$ and $\beta_j \sim f_\beta$, where f_α and f_β are distinct *pdf*'s. As assumed in the previous sections, α_j and β_j are unobservable until the arrival of the j^{th} task and take on new values with each task arrival. The goal, as in previous sections, is to assign the arriving tasks to the workers with values α and β so as to maximize the expected total reward. The difficulty in performing assignments in Model IV.1 arises since both task and worker values are random. Moreover, worker values are not assumed to be identically distributed anymore, and hence, no fixed ranking of worker values exist in advance. Recall from (5.1) that $a_{1,2} := E[X]$, $a_{1,3} := E[X \wedge a_{1,2}]$, and $a_{2,3} := E[X \vee a_{1,2}]$. For an arbitrary random variable γ, let $a_{i,n}^{(\gamma)}$ denote the breakpoint $a_{i,n}$ defined by (5.1) in [12], under the

assumption that the sequentially-arriving task values are given by γX (instead of X). Theorem 20 presents the optimal assignment policy for Model IV.1.

Theorem 20. *The optimal decision, for Model IV.1, upon the arrival of the first task is to assign it to the worker with value β if and only if*

$$W_1^{(\beta-\alpha)} \geq a_{1,2}^{(\beta-\alpha)}, \tag{5.19}$$

where $W^{(\gamma)} := \gamma X$, for any random variable γ. The optimal expected total reward obtained under this policy is given by

$$V^{(\alpha,\beta)} := V_1^{(2)} = 2\, a_{1,2}^{(\alpha)} + a_{2,3}^{(\beta-\alpha)}. \tag{5.20}$$

Proof. Observe that

$$V_1^{(2)}(x) = \max\left\{\alpha_1 x + E\left[W^{(\beta)}\right],\, \beta_1 x + E\left[W^{(\alpha)}\right]\right\},$$

and hence, it is optimal to choose the worker with value β upon the arrival of the first task if and only if $\beta_1 x + E\left[W^{(\alpha)}\right] \geq \alpha_1 x + E\left[W^{(\beta)}\right]$, which is equivalent to (5.19). To compute the expected total reward, let F denote the cumulative distribution function of $W^{(\beta-\alpha)}$, with $\bar{F} := 1 - F$, and note that

$$\begin{aligned}
V_1^{(2)} &= F\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)} + E\left[W^{(\beta)}\right] | W^{(\beta-\alpha)} \leq a_{1,2}^{(\beta-\alpha)}\right] \\
&\quad + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\beta)} + E\left[W^{(\alpha)}\right] | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] \\
&= F\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\beta)}\right] + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)}\right] + F\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)} | W^{(\beta-\alpha)} \leq a_{1,2}^{(\beta-\alpha)}\right] \\
&\quad + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\beta-\alpha)} | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)} | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] \\
&= F\left(a_{1,2}^{(\beta-\alpha)}\right) a_{1,2}^{(\beta-\alpha)} + E\left[W^{(\alpha)}\right] + F\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)} | W^{(\beta-\alpha)} \leq a_{1,2}^{(\beta-\alpha)}\right] \\
&\quad + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\alpha)} | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\beta-\alpha)} | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] \\
&= 2E\left[W^{(\alpha)}\right] + F\left(a_{1,2}^{(\beta-\alpha)}\right) a_{1,2}^{(\beta-\alpha)} + \bar{F}\left(a_{1,2}^{(\beta-\alpha)}\right) E\left[W^{(\beta-\alpha)} | W^{(\beta-\alpha)} > a_{1,2}^{(\beta-\alpha)}\right] \\
&= 2E\left[W^{(\alpha)}\right] + E\left[W^{(\beta-\alpha)} \vee a_{1,2}^{(\beta-\alpha)}\right] \\
&= 2\, a_{1,2}^{(\alpha)} + a_{2,3}^{(\beta-\alpha)}.
\end{aligned}$$

□

Now, consider the following SSAP (labeled S1). Two tasks arrive sequentially to be assigned to the following two workers: a worker with a random success rate of $\beta - \alpha$ and a worker with value zero. To obtain the maximum expected total reward, X_1 is matched with $\beta - \alpha$ if and only if $W_1^{(\beta-\alpha)} \geq E\left[W^{(\beta-\alpha)}\right]$, which is similar to (5.19). This motivates us to break Model IV.1 into two separate SSAP's. The first SSAP is S1,

and the second SSAP (labeled S2) is a problem with two tasks and two identical workers with random success rate α. Suppose that the stream of arriving tasks is the same for S1 and S2. Specifically, when the first task with value $X_1 = x_1$ arrives, it is automatically assigned to the worker with success rate α in S2 (since workers are identical). For S1, $X_1 = x_1$ is matched with $\beta - \alpha$ if (5.19) is satisfied; otherwise, it is assigned to the worker with success rate zero (i.e., rejected). When the second task with value $X_2 = x_2$ is observed, it is again assigned to the worker with value α in S2. For S1, $X_2 = x_2$ is matched with the last remaining worker (with value $\beta - \alpha$ or zero, depending on the decision made at time period one). The optimal expected total reward is equal to $E\left[W^{(\beta-\alpha)} \vee E\left[W^{(\beta-\alpha)}\right]\right]$ and $2E\left[W^{(\alpha)}\right]$ for S1 and S2, respectively. Therefore, solving S1 and S2 simultaneously yields an overall optimal expected total reward equal to that of Model IV.1, as denoted in (5.20). In other words, one can break Model IV.1 into two simpler SSAP's, S1 and S2 (Recall that the optimal assignment policy for S1 is the same as that of Model IV.1).

Model IV.2 extends this setting to a problem of size $n = 3$, where three tasks arrive sequentially to be assigned to two workers with random success rates α and β and one worker with fixed value zero (assigning a task to this worker is equivalent to rejecting that task). We define a new set of breakpoints as follows, which characterize the optimal assignment policy for Model IV.2 in Theorem 21:

$$\begin{aligned}
\tilde{a}_{1,3}^{(\alpha)} &:= a_{1,3}^{(\beta)} - a_{1,3}^{(\beta-\alpha)} \\
\tilde{a}_{1,3}^{(\beta)} &:= a_{1,3}^{(\alpha)} - a_{1,3}^{(\alpha-\beta)} \\
\tilde{a}_{2,3}^{(\alpha-\beta)} &:= a_{2,3}^{(\alpha)} - a_{2,3}^{(\beta)}
\end{aligned}$$

Theorem 21. *Upon the arrival of X_1 in model IV.2, it is optimal to assign the first task with value X_1 to the worker with success rate $l \in \{\alpha, \beta, 0\}$, if event $\mathcal{C}_l$ happens, where*

$$\begin{aligned}
\mathcal{C}_\alpha &:= \left\{W^{(\alpha)} > \tilde{a}_{1,3}^{(\alpha)},\, W^{(\alpha-\beta)} > \tilde{a}_{2,3}^{(\alpha-\beta)}\right\}, \\
\mathcal{C}_\beta &:= \left\{W^{(\beta)} > \tilde{a}_{1,3}^{(\beta)},\, W^{(\alpha-\beta)} \leq \tilde{a}_{2,3}^{(\alpha-\beta)}\right\}, \\
\mathcal{C}_0 &:= \left\{W^{(\alpha)} \leq \tilde{a}_{1,3}^{(\alpha)},\, W^{(\beta)} \leq \tilde{a}_{1,3}^{(\beta)}\right\},
\end{aligned} \tag{5.21}$$

and the resulting optimal expected total reward is given by

$$V_1^{(3)} = \left(E\left[W^{(\alpha)}|\mathcal{C}_\alpha\right] + V^{(0,\beta)}\right) P\{\mathcal{C}_\alpha\} + \left(E\left[W^{(\beta)}|\mathcal{C}_\beta\right] + V^{(0,\alpha)}\right) P\{\mathcal{C}_\beta\} + V^{(\alpha,\beta)} P\{\mathcal{C}_0\}. \tag{5.22}$$

Proof. Observe that if we opt to assign the first task with value $X_1 = x$ to the worker with success rate α, then we are left with two workers (with success rates β and zero) and two arriving tasks. The optimal expected total reward for this problem over the last two remaining time periods is $a_{2,3}^{(\beta)} = E\left[W^{(\beta)} \vee E\left[W^{(\beta)}\right]\right]$.

Comparing this value with the expression in (5.20) reveals that it is equal to $V^{(0,\beta)}$. A similar argument can be made for the other two cases where X_1 is assigned to worker β or to the worker with value zero. It follows that:

$$V_1^{(3)}(x) = \max\left\{\alpha x + V^{(0,\beta)}, \beta x + V^{(0,\alpha)}, V^{(\alpha,\beta)}\right\}$$

It is optimal to reject the first task with value X_1 (i.e., to assign it to the worker with value zero) if $V^{(\alpha,\beta)} \geq \alpha x + V^{(0,\beta)}$ and $V^{(\alpha,\beta)} \geq \beta x + V^{(0,\alpha)}$, which are equivalent to the occurrence of the event $\mathcal{C}_0$. To see why this is the case, note that an argument analogous to that presented in Theorem 20 leads to

$$V^{(\alpha,\beta)} = 2a_{1,2}^{(\beta)} + a_{2,3}^{(\alpha-\beta)}, \tag{5.23}$$

where $2a_{1,2}^{(\beta)} = a_{1,3}^{(\beta)} + a_{2,3}^{(\beta)}$ and $a_{2,3}^{(\alpha-\beta)} = -a_{1,3}^{(\beta-\alpha)}$. Therefore, $V^{(\alpha,\beta)} \geq \alpha x + V^{(0,\beta)}$ is equivalent to $W^{(\alpha)} \leq \tilde{a}_{1,3}^{(\alpha)}$. In a similar fashion, it is proven that $V^{(\alpha,\beta)} \geq \beta x + V^{(0,\alpha)}$ is the same as $W^{(\beta)} \leq \tilde{a}_{1,3}^{(\beta)}$.
It is optimal to assign the first task with value X_1 to the worker with value α if

$$\begin{aligned} &\alpha x + V^{(0,\beta)} \geq V^{(\alpha,\beta)}, \\ &\alpha x + V^{(0,\beta)} \geq \beta x + V^{(0,\alpha)}, \end{aligned} \tag{5.24}$$

or equivalently,

$$\begin{aligned} &\alpha x \geq V^{(\alpha,\beta)} - V^{(0,\beta)}, \\ &(\alpha - \beta)x \geq V^{(0,\alpha)} - V^{(0,\beta)}. \end{aligned} \tag{5.25}$$

From the argument made for the first case (i.e., rejecting X_1), it follows that $V^{(\alpha,\beta)} = a_{1,3}^{(\beta)} + a_{2,3}^{(\beta)} - a_{1,3}^{(\beta-\alpha)}$. Therefore, the first inequality in (5.25) simplifies to

$$\begin{aligned} \alpha x &\geq V^{(\alpha,\beta)} - V^{(0,\beta)} \\ &= \left(a_{1,3}^{(\beta)} + a_{2,3}^{(\beta)} - a_{1,3}^{(\beta-\alpha)}\right) - a_{2,3}^{(\beta)} \\ &= a_{1,3}^{(\beta)} - a_{1,3}^{(\beta-\alpha)} \\ &= \tilde{a}_{1,3}^{(\alpha)}, \end{aligned} \tag{5.26}$$

and the second inequality in (5.25) is re-written as

$$(\alpha - \beta)x \geq V^{(0,\alpha)} - V^{(0,\beta)} = a_{2,3}^{(\alpha)} - a_{2,3}^{(\beta)} = \tilde{a}_{2,3}^{(\alpha-\beta)}. \tag{5.27}$$

Note that (5.26) and (5.27) are equivalent to the event $\mathcal{C}_\alpha$, which implies that it is optimal to pick the

worker with value α upon the arrival of the first task if the event $\mathcal{C}_\alpha$ occurs. The rest of the proof follows along the same lines. $\square$

Unlike Model IV.1 and judging from the optimal policy presented in Theorem 21, Model IV.2 cannot be solved by breaking the problem down into a set of smaller SSAP's. Specifically, the points characterizing the events $\mathcal{C}_\alpha$, $\mathcal{C}_\beta$, and $\mathcal{C}_0$ are functions of the breakpoints defined by (5.1) in [12], and computing these points involve solving stochastic sequential assignment problems for task values given by αX, βX, and $(\alpha - \beta)X$. However, $\tilde{a}_{1,3}^{(\alpha)}$, $\tilde{a}_{1,3}^{(\beta)}$, and $\tilde{a}_{2,3}^{(\alpha-\beta)}$ are not equal to the $\{a_{i,n}\}$'s calculated for SSAP's of a smaller size. Another issue that differentiates the optimal policy of this problem from that of a SSAP with deterministic success rates is that the ordering of the points $\tilde{a}_{1,3}^{(\alpha)}$, $\tilde{a}_{1,3}^{(\beta)}$, and $\tilde{a}_{2,3}^{(\alpha-\beta)}$ is dependent on the specific task and worker vale distributions, and hence, varies from one problem to another, as opposed to the sequence $\{a_{i,n}\}$ that only depends on the task distributions and is proven to be a non-decreasing function of i. One can solve problems of size $n \geq 3$ using a similar approach as that demonstrated by Theorem 21; however, no specific pattern can be found which relates the optimal breakpoints of a problem of size n to that of a problem with size $n+1$. Therefore, the problem can also be modelled as an MDP and solved numerically using backward optimality equations.

5.3 Conclusion

The existing SSAP literature studies variations of the problem under the general assumption that worker values are deterministic and fixed numbers. Chapter 5 extends the SSAP model to the case where worker success rates are observable random variables, taking on new values upon each task arrival. Various models of SSAP with random success rates are studied with different assumptions regarding the success rates' distribution and their dependency on one another. Optimal assignment policies that maximize the expected total reward are presented for each model. These models are categorized into four classes as follows: (1) IID success rates in a given time period, where the vector of success rates are independent from one another through time, (2) an exchangeable set of success rates at a given time period, with worker success rates at a given time being dependent on that of preceeding time periods, (3) time-dependent random success rates which form two distinct worker classes (based on their values), with one class containing workers with value zero, and (4) time-dependent random success rates with distinct generic distributions, forming two or more than two worker classes.

Further research is required to address more general version of the SSAP with random success rates. One challenge is to study this problem, where task arrival times are no longer deterministic; e.g., tasks arriving according to a Poisson process. Another research direction is the case where worker values are updated at

each task arrival according to the realized task value (i.e., worker and task values are dependent on one another). Studying this problem under risk sensitive objective functions, such as the threshold criteria, is another possible extension.

Chapter 6

Discussion

The stochastic sequential assignment problem (SSAP) considers how to allocate N available distinct workers to N independently and identically distributed (IID) sequentially arriving tasks with stochastic parameters, such that the expected total reward obtained from the sequential assignments is maximized. This thesis focuses on studying practical variations and extensions of the SSAP, with the goal of eliminating restricting assumptions so that the problem setting converges to that of real-world problems. Some possible extensions to the methods and results presented in this dissertation is now presented.

As shown in Chapter 5, when studying SSAP's with random success rates, the most difficult issue that arises is dealing with the case that worker values are not identically distributed and no ranking of their values exist in advance. Class IV models circumvent this issue for a SSAP with two worker categories of distinct distributions (and non-zero values) by breaking the parent problem down into two separate subproblems, S1 and S2. Complexity and stochasticity in eash of these subproblems are decreased compared to the parent problem, in the sense that closed form expressions for optimal policies of S1 and S2 are already available and provided by the results in Class III models. Solving S1 and S2 individually upon each task arrival nicely yields the optimal assignment decision for the parent problem at each stage. However, this result cannot be extended to SSAP's with more than two distincly distributed worker categories. This is demonstrated by solving and providing a closed form expression for optimal policy for the case with three worker categories. As elaborated in Chapter 5, the best approach known so far to tackle this class of problems (with more than three worker categories) is for them to be modelled as a MDP's and solved numerically. Hence, a possible future research direction is to find a way around this issue so that the optimal assignment policy is presented in closed form.

Another intriguing aspect of the SSAP to analyze is to consider different arrival time distributions for tasks. Specifically, one can study the variations of SSAP introduced in this dissertation, under the more realistic assumption that task arrival times are not deterministic or known in advance. Moreover, as all the four classes of models in Chapter 5 solve the problem under the risk-neutral objective function of maximizing the expected total reward, all these models need to be evaluated and studied from the threshold criteria

point of view or other risk-sensitive objective function, to adjust to a range of standards from different decision-makers.

Appendix A

Proof of Proposition 2. Fix $\tilde{\epsilon}$ such that $0 < \tilde{\epsilon} < \frac{\epsilon}{\pi_{i-1}-\pi_i}$, then there exists $\tilde{n} \in \mathbb{N}$ such that

$$\frac{[n(1-\epsilon)]}{r_i} \in (\alpha - \tilde{\epsilon}, \alpha + \tilde{\epsilon}),$$

for $n \geq \tilde{n}$, where $\alpha := \frac{1-\epsilon}{\pi_{i-1}-\pi_i}$. Therefore, for $n \geq \tilde{n}$,

$$\begin{aligned} P\left\{\frac{U_{r_i}^{(n)}}{r_i} \leq \frac{[n(1-\epsilon)]}{r_i}\right\} &\leq P\left\{\frac{U_{r_i}^{(n)}}{r_i} < \alpha + \tilde{\epsilon}\right\} \\ &= P\left\{\frac{U_{r_i}^{(n)}}{r_i} - \frac{1}{\pi_{i-1}-\pi_i} < -\left(\frac{\epsilon}{\pi_{i-1}-\pi_i} - \tilde{\epsilon}\right)\right\} \\ &\leq P\left\{\left|\frac{U_{r_i}^{(n)}}{r_i} - \frac{1}{\pi_{i-1}-\pi_i}\right| > \frac{\epsilon}{\pi_{i-1}-\pi_i} - \tilde{\epsilon}\right\} \\ &\to 0 \quad as \quad n \to +\infty, \end{aligned}$$

since $\frac{U_{r_i}^{(n)}}{r_i} \to \frac{1}{\pi_{i-1}-\pi_i}$ in probability as $n \to +\infty$.

Proof of Theorem 5. Fix an arbitrary $\epsilon > 0$, and observe that

$$\begin{aligned} P\left\{\left|\frac{R_n}{n} - r^*\right| > \epsilon\right\} &= P\left\{\left|\left(\frac{R_n^{(1)}}{n} - r^*\right) + \frac{R_n^{(2)}}{n}\right| > \epsilon\right\} \\ &\leq P\left\{\left|\frac{R_n^{(1)}}{n} - r^*\right| > \frac{\epsilon}{2}\right\} + P\left\{\left|\frac{R_n^{(2)}}{n}\right| > \frac{\epsilon}{2}\right\}. \end{aligned} \tag{A.1}$$

Note that

$$\begin{aligned} P\left\{\left|\frac{R_n^{(1)}}{n} - r^*\right| > \frac{\epsilon}{2}\right\} &= P\left\{\left|\frac{1}{n}\sum_{i=1}^{U^{(n)}}\sum_{j=1}^{k} p_j X_i I_{\{X_i \in \mathcal{A}_j\}} - r^*\right| > \frac{\epsilon}{2}\right\} \\ &\leq P\left\{\sum_{j=1}^{k}\left|\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_j\}} - E\left[X_i I_{\{X_i \in \mathcal{A}_j\}}\right]\right| > \frac{\epsilon}{2p_1}\right\} \\ &\leq \sum_{j=1}^{k} P\left\{\left|\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i \in \mathcal{A}_j\}} - E\left[X_i I_{\{X_i \in \mathcal{A}_j\}}\right]\right| > \frac{\epsilon}{2kp_1}\right\}. \end{aligned}$$

Now, pick any j such that $1 \leq j \leq k$, and observe that

$$\begin{aligned} P\left\{\left|\frac{1}{n}\sum_{i=1}^{U^{(n)}} X_i I_{\{X_i\in\mathcal{A}_j\}} - E\left[X_i I_{\{X_i\in\mathcal{A}_j\}}\right]\right| > \frac{\epsilon}{2kp_1}\right\} &\leq P\left\{\left|\frac{1}{n}\sum_{i=1}^{n} X_i I_{\{X_i\in\mathcal{A}_j\}} - E\left[X_i I_{\{X_i\in\mathcal{A}_j\}}\right]\right| > \frac{\epsilon}{4kp_1}\right\} \\ &\quad + P\left\{\left(\frac{1}{n}\sum_{i=U^{(n)}+1}^{n} X_i I_{\{X_i\in\mathcal{A}_j\}}\right) I_{\{U^{(n)}<n\}} > \frac{\epsilon}{4kp_1}\right\} \\ &\leq P\left\{\left|\frac{1}{n}\sum_{i=1}^{n} X_i I_{\{X_i\in\mathcal{A}_j\}} - E\left[X_i I_{\{X_i\in\mathcal{A}_j\}}\right]\right| > \frac{\epsilon}{4kp_1}\right\} \\ &\quad + P\left\{\left(1 - \frac{U^{(n)}}{n}\right) > \frac{\epsilon}{4Mkp_1}\right\}, \end{aligned} \tag{A.2}$$

where the first inequality in (A.2) follows from (3.6). As the number of tasks increases, by the weak law of large numbers, the first term on the right-hand side of (A.2) can be made arbitrarily small. With an argument similar to that made in the proof of Proposition 2, it can be concluded that the second term on the right-hand side of (A.2) also approaches zero. Hence, the left-hand side of (A.2) converges to zero as $n \to +\infty$ for any $1 \leq j \leq k$, which implies that

$$\lim_{n\to+\infty} P\left\{\left|\frac{R_n^{(1)}}{n} - r^*\right| > \frac{\epsilon}{2}\right\} = 0.$$

Next, the second term on the right-hand side of (A.1) is proved to converge to zero as follows:

$$\begin{aligned} \lim_{n\to+\infty} P\left\{\left|\frac{R_n^{(2)}}{n}\right| > \frac{\epsilon}{2}\right\} &\leq \lim_{n\to+\infty} P\left\{p_1\left(\sum_{i=U^{(n)}+1}^{n} X_i\right) I_{\{U^{(n)}<n\}} > \frac{\epsilon}{2}\right\} \\ &\leq \lim_{n\to+\infty} P\left\{\left(1 - \frac{U^{(n)}}{n}\right) > \frac{\epsilon}{2p_1 M}\right\} \\ &= 0, \end{aligned}$$

and hence, $\lim_{n\to+\infty} P\left\{\left|\frac{R_n}{n} - r^*\right| > \epsilon\right\} = 0$.

Appendix B

Stochastic Sequential Assignment Problem with Dependency and Random Number of Tasks

B.1 Introduction

Consider the stochastic sequential assignment problem (SSAP) introduced in [12], where there are N workers available to perform N tasks. The tasks occur in sequential order with independent and identically distributed (IID) random variables X_j, denoting the value of the jth task. In addition, a fixed probability p_i (success rate) is associated with each worker. Whenever the ith worker is assigned to the jth task, the worker becomes unavailable for future assignments, with the expected reward associated with this assignment given by $p_i x_j$, where x_j is the observed value of the jth task. The objective is to assign the N workers to N tasks so as to maximize the expected total reward. It is shown in [12] that there exists real numbers

$$-\infty = a_{0,N} \leq a_{1,N} \leq a_{2,N} \leq \cdots \leq a_{N,N} = +\infty, \tag{B.1}$$

such that the optimal choice in the initial stage is to assign the ith best available worker if the random variable X_1 falls within the ith highest interval (i.e., $x_1 \in (a_{N-i,N}, a_{N-i+1,N}]$). Furthermore, $a_{i,N}$ is the expected value of the quantity assigned to the worker with the ith smallest success value in an $(N-1)$-stage problem and depends only on the distribution of the $\{X_j\}$.

This dissertation studies an extension of SSAP, where the task value in each time period depends on the value of the preceding task. Note that this problem contains as a special case an SSAP in which the dependency between task values are governed by a Markov chain. Also, it is assumed here that a task arrives only with a certain probability in each time period (independent of the other time periods), and hence, the total number of tasks is a random variable following a Binomial distribution. The objective function is to maximize the expected total reward as in [12], and an optimal assignment policy is established which is of the same form as that proposed for the original SSAP by [12] with the difference that the interval breakpoints are random variables instead of deterministic fixed numbers. Furthurmore, a generalization of this problem is studied where the total number of arriving tasks is unknown until after the final arrival and is a random variable following an arbitrary probability mass function (pmf) with finite support. This problem is further

extended to the case where the *pmf* of the number of tasks has infinite support, and an optimal policy is proposed so as to achieve the maximum expected total reward in the infinite-horizon problem.

The present work deviates from the existing SSAP literature by combining dependency between task values with randomness in the number of tasks. In fact, only three papers can be found in literature (more precisely, [4], [17], and [21]) that have incorporated dependency into SSAP. Furthermore, in all these papers, it is assumed that the total number of tasks to arrive is a given deterministic number and known apriori. Likewise, papers focusing on variations of SSAP with random number of tasks (specifically, [2], [3], [22], and [28]) have all considered the task values to be independent of each other. There is an absence of work on the SSAP where dependent task values and random number of tasks are incorporated simultaneously into the model, and hence, this thesis addresses a problem in which the number of tasks is assumed to follow an arbitrary probability mass function with either finite or infinite support while the task value in each time period depends on the value of the preceding task. The dependency modelled here also has the Markov-modulated dependencies introduced in [4] and [21] (with a finite-state Markov chain governing task values) as a special case. Moreover, the constraining assumptions in [21] (e.g., those imposed on elements of the transition probability matrix) are relaxed in the present work with the only requirement being that the task values are integrable.

There are several extensions to the investigations by [12]. Some works (namely, [4] and [21]) have introduced a Markov chain approach to the SSAP. In [4], the Secretary Problem is studied where the best k out of N sequentially arriving secretaries must be hired. The total number of secretaries is fixed and known apriori, and the values of the successive secretaries are random variables from r different known distributions, where the successive distributions are governed by a Markov chain. It is shown in [4] that r separate sets of $a_{i,N}$'s, one for each distribution type, should be computed recursively at each time period, and these breakpoints are dependent on the breakpoints calculated for all the other secretary types during all previous time periods. [21] considers SSAP over a finite-state Markov chain in which the states are not known explicitly, but their transition probability matrix is given. A non-negative random variable is associated with each state of the process with the relationship between each state of the Markov chain and the observed random variable being known. Observing a realization of these N sequentially arriving random variables (with N known beforehand), the decision-maker may choose from a finite set of actions in each time period so as to maximize the total expected reward. In order for the results to hold, constraining assumptions are made about the elements of the transition probability matrix and also on the *cdf* of the random variables. Apart from [4] and [21], the only other paper that considers dependent task values in SSAP is [17], which does so without defining a Markov chain. More specifically, assuming that the number of tasks is deterministic and known in advance, an optimal assignment policy is established to maximize the

expected total reward in [17]. An infinite-horizon SSAP (i.e., an SSAP with an infinite number of arriving tasks) is also considered in [17], and an optimal assignment policy is proposed.

The other category of papers address an extension of SSAP focusing on the number of arriving tasks or their arrival-time distributions. [2] analyzes a problem where N IID tasks with distribution F arrive sequentially at random times according to a renewal process and must be assigned to a fixed set of workers. The objective function is to maximize the expected reward per unit time. The same problem is studied by [3], where various arrival-time distributions are considered. [28] also examines SSAP where tasks arrive according to a Poisson process and are assigned random deadlines. [22] considers a variation of SSAP in which the number of tasks is unknown until after the final arrival and follows a given probability distribution; however, the arriving tasks are assumed to be independent of each other (but not necessarily identically distributed).

Numerous other variations and applications of SSAP have been studied. An application of SSAP in kidney allocation to patients has been addressed in [31]. [13] have modeled an investment problem using SSAP. [19] and [23] have addressed applications of SSAP in aviation security. Moreover, asymptotic behavior of the Secretary Problem has been analyzed in [7], and an optimal policy has been established as the number of arriving tasks approaches infinity. Finally, [6] and [15] consider variations of SSAP under incomplete information.

This part of the dissertation is organized as follows. The next section provides illustrative examples and applications of our model. Section 3 presents the mathematical model of SSAP (for both finite and infinite N) with dependent task values where a task arrives with probability α during each period (resulting in the number of tasks to follow a Binomial distribution) and characterizes the optimal policy to maximize the expected total reward. The fourth section studies an extension of this problem in which the number of arriving tasks is unknown until after the final arrival and follows an arbitrary probability mass function. Section 5 presents the numerical results. The last section offers concluding comments and future directions of research.

B.2 Illustrative Examples and Applications

This section provides examples, which demonstrate the application of SSAP with dependency between task values. The SSAP in its basic form assumes tasks to be independently and identically distributed, which is not always a reasonable assumption for real-world problems. For example, consider the case where task values represent the worth of arriving tasks and the task worth at time period j is denoted by X_j. It is safe to assume that the task worth is dependent on the economic conditions upon the arrival of that task ([21]). Suppose that the changes in these conditions are modeled by a Markov chain with state space $\tilde{S}$, where the states of the chain represent the economic conditions. Due to the tasks' dependence on the economic conditions

and also the dependency within these conditions (through the Markov chain transitions), if follows that the consecutive task values are dependent on each other. To elaborately model this statement in mathematical terms, we assume that if the economic condition upon the arrival of X_j is given by $k \in \tilde{S}$, then X_j is derived from the distribution function F_k. Therefore, the successive task distributions are governed by the same Markov chain, leading to the dependency between task values. Once the task worth X_j is observed, one needs to choose a worker from the set of available resources to assign to X_j and to repeat this procedure for all the arriving tasks so as to maximize the expected total reward.

Another example is the sequential allocation model introduced by [14], where there are D units available for investment, and an investment opportunity arises during each of the N time periods. Observing the investment return rate in a given time period, the decision-maker must decide on how many units to invest during that period. The number of units to invest and the investment return rate during a time period correspond to the worker value and the task value in that period, respectively. The objective function is to maximize the expected total return from all the investments made. [14] assume the task values (or equivalently, the return rates) to be independently and identically distributed, however; we can drop this assumption to obtain a more realistic model, with dependent task values. Similar to the situation explained above, the investment return rates are functions of economic conditions, which vary from time to time. This results in the successive return rates to be dependent on each other, as the transition from one economic condition to the other occurs.

The last example is an application of SSAP in aviation security, where sequentially arriving passengers are assigned to available security resources, as they check in at an airport. Specifically, assume that the time interval for screening passengers is divided into N slots (stages), where passenger j arrives during stage j. Upon the arrival of each passenger, a pre-screening system determines their threat value, classifying them as non-selectees (i.e., the passengers who have been cleared of posing a risk) or selectees (i.e., those who have not been cleared, based on available information known about them [19]). Each assessed threat value is defined as the probability that a passenger carries a threat item, with X_j indicating the threat value of passenger j. The capacity of the selectee class (i.e., the number of available screening devices associated with the selectee class) is c, and N denotes the capacity of the non-selectee class. Define the security level to be the conditional probability of detecting a passenger with a threat item given that they are classified as selectees or non-selectees, and let L_S and L_{NS} be the security levels associated with the selectee and non-selectee classes. Moreover, let $\gamma_j = 1$ and $\gamma_j = 0$ denote the j^{th} passenger assignment as a selectee and

a non-selectee, respectively. The *total security* for this setting is defined as

$$\sum_{j=1}^{n} X_j \left[L_S \gamma_j + L_{NS}(1 - \gamma_j) \right],$$

where the objective is to find a policy for assigning passengers to classes as they check in so as to maximize the expected total security. If the passengers, planning an attack, work in groups of two or more people, then the passenger threat values (or equivalently, task values) depend on one another in the sense that if one passenger is classified as a high-risk passenger, then there is a high (or low, depending on the attack strategy) probability that the person standing right behind them in the line is also a threat to the safety of the airport and other passengers.

Section 3 presents the mathematical model of SSAP with dependent task values where the number of tasks follow a Binomial distribution and characterizes the optimal policy to maximize the expected total reward.

B.3 The Model

This section addresses an extension of SSAP, where a task arrives with probability α in each time period and the values of any two consecutive tasks are dependent on each other. An SSAP in which the dependency between task values is governed by a Markov chain can be viewed as a special case of this problem. An optimal policy is established to maximize the expected total reward obtained from assigning the sequentially arriving tasks to available workers.

Suppose that there are N workers available to sequentially perform tasks during N time periods, with a value p_i associated with the ith worker, where without loss of generality $p_1 \geq p_2 \geq \cdots \geq p_N$. The p_i's are the success rates of the workers, where the larger the success rate, the better the worker. A task arrives with probability α during each time period, independently of other time periods, with at most one task arriving during a time period. This is a reasonable assumption given that the length of each time period can be set arbitrarily small. Suppose that X_j takes values in the set $\mathcal{S} \cup \{0\}$, where $\mathcal{S} \subseteq (0, +\infty)$, and X_j is integrable on $\mathcal{S}$. Assume that when a task actually arrives at time period j, its value is positive (or equivalently, $X_j \in \mathcal{S}$), while $X_j = 0$ denotes the case where no task arrives during the jth time period (which occurs with probability $1 - \alpha$). Therefore, the number of tasks that arrive over the N time periods is a Binomial random variable with parameters (N, α) where the value of any two successive tasks are dependent on each other; specifically, if a task arrives during the jth time period, then its value depends on X_{j-1}. For $0 < \alpha \leq 1$, the

dependency between any two successive task values is given by

$$\begin{aligned} g_\alpha(0 \mid y) &:= P_\alpha \{X_{n+1} = 0 \mid X_n = y\}, \\ g_\alpha(\mathcal{B} \mid y) &:= P_\alpha \{X_{n+1} \in \mathcal{B} \mid X_n = y\}, \end{aligned} \tag{B.2}$$

for $1 \leq n \leq N-1$, $y \in \mathcal{S} \cup \{0\}$, and $\mathcal{B} \subseteq \mathcal{S}$, where P_α implies that the conditional probability is computed given that a task arrives with probability α during a time period, and hence, it follows that

$$\begin{aligned} g_\alpha(\mathcal{S} \mid y) &= \alpha, \\ g_\alpha(0 \mid y) &= 1 - \alpha, \end{aligned} \tag{B.3}$$

for $y \in \mathcal{S} \cup \{0\}$. Moreover, $\bar{g}_\alpha$ denotes the probability distribution of the initial task with

$$\begin{aligned} \bar{g}_\alpha(0) : &= P_\alpha \{X_1 = 0\} \\ &= 1 - \alpha, \end{aligned} \tag{B.4}$$

and

$$\bar{g}_\alpha(\mathcal{B}) := P_\alpha \{X_1 \in \mathcal{B}\}, \tag{B.5}$$

for all $\mathcal{B} \subseteq \mathcal{S}$ such that $\bar{g}_\alpha(\mathcal{S}) = \alpha$.

It will be proven in Theorem 22 that the optimal policy to maximize the expected total reward is qualitatively of the same form as that obtained in [12]. The proof technique applies Lemma 16 (due to [16]) to obtain the structure of the optimal policy.

Lemma 16. *(Hardy's Theorem) If $x_1 \leq x_2 \leq ... \leq x_n$ and $y_1 \leq y_2 \leq ... \leq y_n$ are sequences of numbers, then*

$$max_{(i_1,i_2,...,i_n) \in V} \sum_{j=1}^{n} x_{i_j} y_j = \sum_{j=1}^{n} x_j y_j, \tag{B.6}$$

where V is the set of all permutations of the integers (1,2,...,n).

Hardy's theorem studies a deterministic version of SSAP, where all task values are fixed and observable at the beginning. It implies that the maximum sum is achieved when the smallest of the x's and y's are paired, the second smallest of the x's and y's are paired, and so forth until the largest of the x's and y's are paired. Intuitively, it follows from this theorem that we need to obtain a ranking of task values to arrive (or a ranking of their expected values in our case, due to sequential arrivals and stochasticity) and match the ith best task, according to this ranking, to the ith best available worker. Specifically, Theorem 22 proves that under the optimal assignment policy and upon the arrival of the nth task, the real line is partitioned

into $N-n+1$ intervals with random breakpoints. By observing the value of X_n, these breakpoints are computed as explained in (B.7)-(B.13), and if X_n falls within the ith highest of these intervals, then it must be assigned to the ith best remaining worker (i.e., the one with the ith highest success rate) to achieve the optimal expected total reward. These breakpoints, which the real line is partitioned into, characterize the optimal policy and are computed recursively by (B.7)-(B.13) as follows (an intiutive interpretation is given right afterwards): For $n = 1, 2, \ldots, N$ and $i > N-n+1$, define

$$Z_{i,n}^{(1)} = -\infty, Z_{i,n}^{(2)} = -\infty, Z_{0,n}^{(1)} = +\infty, Z_{0,n}^{(2)} = +\infty. \tag{B.7}$$

Also, let

$$Z_{N-n+1,n}^{(1)} = \alpha \left(X_n \vee E[Z_{N-n+1,n+1}^{(1)} \mid X_{n+1} > 0,\, X_n] \right) \wedge E[Z_{N-n,n+1}^{(1)} \mid X_{n+1} > 0,\, X_n], \tag{B.8}$$

$$Z_{N-n+1,n}^{(2)} = 0, \tag{B.9}$$

and

$$\begin{aligned} Z_{m,n}^{(1)} = \alpha \Big(X_n \vee \Big(E[Z_{m,n+1}^{(1)} \mid X_{n+1} > 0,\, X_n] + Z_{m,n+1}^{(2)} \Big) \Big) \wedge \\ \Big(E[Z_{m-1,n+1}^{(1)} \mid X_{n+1} > 0,\, X_n] + Z_{m-1,n+1}^{(2)} \Big), \end{aligned} \tag{B.10}$$

$$Z_{m,n}^{(2)} = (1-\alpha) \left(E[Z_{m,n+1}^{(1)} \mid X_{n+1} > 0,\, X_n = 0] + Z_{m,n+1}^{(2)} \right), \tag{B.11}$$

for any $1 \leq n \leq N-1$ and $m = 1, 2, ..., N-n$, where $\vee$ and $\wedge$ denote the maximum and the minimum, respectively. Note that the expressions given in (B.8)-(B.11) are computed using probability distributions introduced by (B.2)-(B.5). Finally, let

$$Z_{1,N}^{(1)} = \alpha X_N, \tag{B.12}$$

and

$$Z_{1,N}^{(2)} = 0. \tag{B.13}$$

Now, we give an intuitive interpretation of the expressions presented in (B.7)-(B.13). Consider an arbitrary set of real numbers $\mathcal{A} = \{a_i, i = 1, 2, \cdots, k\}$. The notion of *ordered values* of this set refers to a set $\mathcal{B} = \{b_j, j = 1, 2, \cdots, k\}$ such that $b_{k-j+1} = a_{[j]}$ for all $j = 1, 2, \cdots, k$ where $a_{[j]}$ is the jth smallest element of $\mathcal{A}$. Observe that $\left\{ \frac{1}{\alpha} Z_{i,n}^{(1)},\ \text{for all } i = 1, 2, \cdots, N-n+1 \right\}$ is the set of ordered values of

$$\left\{ X_n, \left(E\left[Z_{m,n+1}^{(1)} \middle| X_{n+1} > 0,\, X_n \right] + Z_{m,n+1}^{(2)} \right),\ \text{for all } m = 1, 2, \cdots, N-n \right\}, \tag{B.14}$$

for any $1 \leq n \leq N-1$. As proven in Thereom 22, the elements of the set, (B.14), actually serve a critical role in characterizing the optimal policy upon the arrival of the nth task. Specifically, when the nth task arrives with $1 \leq n \leq N-1$, the real line is partitioned into $N-n+1$ random intervals with endpoints given by

$$-\infty,\ E\left[Z^{(1)}_{N-n,n+1}\middle|X_{n+1}>0,\ X_n\right]+Z^{(2)}_{N-n,n+1},\cdots,\ E\left[Z^{(1)}_{1,n+1}\middle|X_{n+1}>0,\ X_n\right]+Z^{(2)}_{1,n+1},\ +\infty, \quad \text{(B.15)}$$

and as mentioned before, the ith best worker must be assigned to X_n under the optimal policy if X_n belongs to the ith highest interval given by (B.15). On the other hand, Theorem 22 implies that $\frac{1}{\alpha}Z^{(1)}_{i,n}$ is the expected value of the task assigned to the ith best worker available upon the arrival of X_n. In other words and as proven mathematically later in Lemma 19, comparing X_n with the threshold values prescribed by the optimal policy is equivalent to finding the rank of X_n (value of the nth task) in the set of expected values of tasks to arrive during time periods n up to N. For example, if X_n is the largest value in this set, then the worker with the largest success rate is assigned to it (due to Hardy's Theorem), since all the other tasks that are yet to arrive have lower (expected) values compared to the current task, X_n.

Upon the arrival of the Nth task, only one worker remains, which is assigned to X_N regardless of the policy applied during the previous stages; in other words, no breakpoints are required to determine the best decision at this final stage. Hence, in order for the policy at time $n=N$ to conform to the optimal policy during the preceding stages, define the set of breakpoints at time $n=N$ to be empty. Consequently, under the optimal policy and upon the arrival of X_N, the real line is viewed as a single interval $(-\infty,+\infty)$ with no breakpoints. This way, X_N falls trivially within the first highest interval (i.e., $X_N \in (-\infty,+\infty)$) so that the first best worker (which is actually the only worker left) is paired with it according to the optimal policy. Before proving the optimality of this policy in Theorem 22, three preliminary lemmas are discussed. Lemmas 17 and 18 provide interesting properties of $\left\{Z^{(k)}_{i,n}\right\}$ for $k=1,2$, which are applied later in proof of Theorem 22, Lemma 19, and Lemma 20.

Lemma 17. *For $n=1,\ldots,N$, $Z^{(1)}_{j,n}$ and $Z^{(2)}_{j,n}$ are non-increasing functions in $j=1,2,\cdots$.*

Proof. The proof is by backward induction on n and follows from the definition of $\left\{Z^{(1)}_{j,n}\right\}$ and $\left\{Z^{(2)}_{j,n}\right\}$. □

Lemma 18. *For $n=1,\ldots,N$, $Z^{(1)}_{j,n}$ and $Z^{(2)}_{j,n}$ are non-negative functions in $j=0,1,\cdots,N-n+1$.*

Proof. Note that the non-negativity of $Z^{(2)}_{j,n}$ for $n=N$ follows from (B.13). Now, fix an arbitrary $1 \leq n \leq N-1$, and recall that $Z^{(2)}_{N-n+1,n}=0$ by (B.9). In addition, $Z^{(2)}_{j,n}$ is a non-decreasing function in j by Lemma 17. Therefore,

$$0=Z^{(2)}_{N-n+1,n} \leq Z^{(2)}_{N-n,n} \leq Z^{(2)}_{N-n-1,n} \leq \cdots \leq Z^{(2)}_{1,n} \leq Z^{(2)}_{0,n}=+\infty,$$

which is the desired result. For $1 \leq n \leq N$, an argument similar to that applied above for $Z_{j,n}^{(2)}$ shows that the non-negativity of $\left\{Z_{j,n}^{(1)}\right\}$ for $j = 0, 1, \cdots, N-n+1$ comes down to proving that $Z_{N-n+1,n}^{(1)} \geq 0$, which is established by backward induction on n. □

Before proceeding to the main theorem, some notations are needed. Let π^* denote the optimal policy and $\pi^*(i,n)$ represent the index of the ith best remaining worker when the nth task is about to be assigned, given that π^* has been applied for the assignment of the first $n-1$ tasks. For example, suppose that the total number of tasks to arrive is $N = 5$. Given that p_4 has been assigned to X_1 under the optimal policy, $\pi^*(4,2)$ indicates the fourth best (remaining) worker upon arrival of X_2; equivalently, $p_{\pi^*(4,2)} = p_5$. In addition, let $\mathcal{W}_n^{\pi^*} := \left\{p_{\pi^*(1,n)}, p_{\pi^*(2,n)}, \cdots, p_{\pi^*(N-n+1,n)}\right\}$ denote the set of available workers at time period n when π^* has been applied for the assignment of the first $n-1$ tasks, where $\mathcal{W}_1^{\pi^*} = \{p_1, p_2, \cdots, p_N\}$ is actually the set of all N workers available before the arrival of the first task. Also, let

$$\mathcal{W}_{n\setminus i}^{\pi^*} := \left\{p_{\pi^*(1,n)}, p_{\pi^*(2,n)}, \cdots, p_{\pi^*(i-1,n)}, p_{\pi^*(i+1,n)}, \cdots, p_{\pi^*(N-n+1,n)}\right\}$$

be the set of available workers after removing the ith best worker (i.e., the one with the ith highest success rate) from the set $\mathcal{W}_n^{\pi^*}$.

Now, define the notion of *remaining reward* under policy π at time period n as the reward that will be obtained from assigning all the remaining $N-n+1$ workers under policy π, when X_n is about to arrive. In fact, it is analogous to the concept of *cost-to-go* in dynamic programming with the difference that it represents the reward instead of the cost. Let $R^{\pi}(\mathcal{W}_n^{\pi^*})$ be the remaining reward at time period n under policy π when $\mathcal{W}_n^{\pi^*}$ is the set of available workers upon the arrival of X_n. In this way, $R^{\pi^*}(\mathcal{W}_n^{\pi^*})$ is the optimal remaining reward at time period n when the set of available workers is given by $\mathcal{W}_n^{\pi^*}$, and hence, $R^{\pi^*}(\mathcal{W}_1^{\pi^*})$ denotes the optimal total reward obtained after assigning all the N workers. Theorem 22 provides the optimal policy to maximize the expected total reward over N time periods, or equivalently, to obtain $E\left[R^{\pi^*}(\mathcal{W}_1^{\pi^*})\middle|X_1\right]$.

Theorem 22. *Suppose that N workers with success rates $p_1 \geq p_2 \geq \cdots \geq p_N$ are available to perform tasks arriving sequentially over N time periods, where a task with positive value arrives with probability α in each time period, independently of the other time periods, and the reward value of a task depends on the value of the preceding task by* (B.2)-(B.5)*. The optimal policy that maximizes the expected total reward is to assign the nth task to the ith best remaining worker if X_n falls in the ith highest interval defined by the breakpoints*

$$\left\{E\left[Z_{m,n+1}^{(1)}\middle|X_{n+1} > 0,\, X_n\right] + Z_{m,n+1}^{(2)},\ \textit{for all } m = 1, 2, ..., N-n\right\}.$$

This policy implies that if $X_n = 0$ (i.e., if no task arrives during time period n), the worst remaining worker (i.e., the one with the smallest success rate) should be eliminated.

Proof. The proof proceeds by backward induction on n. The induction assumptions are:

- When task n with value X_n arrives, the optimal policy is to assign this task to the ith best remaining worker if X_n falls in the ith highest interval defined by

$$\left\{E\left[Z^{(1)}_{m,n+1}\middle|X_{n+1}>0,\, X_n\right] + Z^{(2)}_{m,n+1},\ \text{for all } m = 1, 2, ..., N-n\right\}.$$

- When X_n arrives (and hence, $N-n+1$ workers remain unassigned), the optimal conditional expected remaining reward is given by

$$E\left[R^{\pi^*}(\mathcal{W}_n^{\pi^*})\middle|X_n\right] = \frac{1}{\alpha}\sum_{i=1}^{N-n+1}\bar{p}_i Z^{(1)}_{i,n}, \tag{B.16}$$

  where $\bar{p}_i$ is the ith largest element of $\mathcal{W}_n^{\pi^*}$.

The first assumption clearly holds true for $n = N$ since as mentioned before, the set of breakpoints at this stage is empty. Therefore, X_N falls trivially within the first highest interval (i.e., $X_N \in (-\infty, +\infty)$), and hence, the first best worker (which is actually the only worker left) is paired with the Nth task. In this case, the optimal conditional expected remaining reward is given by

$$E\left[R^{\pi^*}(\mathcal{W}_N^{\pi^*})\middle|X_N\right] = p_{\pi^*(1,N)}X_N = \frac{1}{\alpha}p_{\pi^*(1,N)}Z^{(1)}_{1,N},$$

where the last equality follows from (B.12). For the induction step, assume that the induction assumptions are true for X_n. It suffices to show that they hold for X_{n-1}. Observe that the optimal conditional expected remaining reward, when the $(n-1)$th task with value x arrives, is given by

$$E\left[R^{\pi^*}(\mathcal{W}_{n-1}^{\pi^*})\middle|X_{n-1}=x\right] = \max_{i=1,2,\cdots,N-n+2}\left\{p_{\pi^*(i,n-1)}x + E\left[R^{\pi^*}(\mathcal{W}_{n-1\setminus i}^{\pi^*})\middle|X_{n-1}=x\right]\right\}. \tag{B.17}$$

Conditioning on whether a task in the next time period (i.e., n) arrives or not, one would obtain

$$\begin{aligned} E\left[R^{\pi^*}(\mathcal{W}_{n-1\setminus i}^{\pi^*})\middle|X_{n-1}=x\right] &= (1-\alpha)E\left[R^{\pi^*}(\mathcal{W}_{n-1\setminus i}^{\pi^*})\middle|X_n=0,\, X_{n-1}=x\right] \\ &\quad + \alpha E\left[R^{\pi^*}(\mathcal{W}_{n-1\setminus i}^{\pi^*})\middle|X_n>0,\, X_{n-1}=x\right]. \end{aligned} \tag{B.18}$$

Note that

$$Z^{(1)}_{N-n+1,n+1} = -\infty \quad \text{and} \quad E\left[Z^{(1)}_{N-n,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] \geq 0,$$

where the first and the second expressions result from (B.7) and Lemma 18, respectively. Now, plug $X_n = 0$ into (B.8) and obtain:

$$\frac{1}{\alpha} Z^{(1)}_{N-n+1,n} = \left(0 \vee E\left[Z^{(1)}_{N-n+1,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right]\right) \wedge E\left[Z^{(1)}_{N-n,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] = 0. \quad \text{(B.19)}$$

Recall that $E\left[Z^{(1)}_{j,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] + Z^{(2)}_{j,n+1} \geq 0$, for any $j = 1, 2, \cdots, N-n$, by Lemma 18. Therefore, plugging $X_n = 0$ into (B.10) results in

$$\begin{aligned} \frac{1}{\alpha} Z^{(1)}_{j,n} &= \left(0 \vee \left(E\left[Z^{(1)}_{j,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] + Z^{(2)}_{j,n+1}\right)\right) \\ &\quad \wedge \left(E\left[Z^{(1)}_{j-1,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] + Z^{(2)}_{j-1,n+1}\right) \\ &= E\left[Z^{(1)}_{j,n+1}\middle| X_{n+1} > 0,\, X_n = 0\right] + Z^{(2)}_{j,n+1}, \end{aligned} \quad \text{(B.20)}$$

where the last equality is the direct result of Lemma 17. Now, let $\bar{p}_j$ denote the jth largest element of the set $\mathcal{W}^{\pi^*}_{n-1\setminus i}$, and observe that

$$\begin{aligned} E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1\setminus i})\middle| X_n = 0,\, X_{n-1} = x\right] &= E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1\setminus i})\middle| X_n = 0\right] \\ &= \frac{1}{\alpha} \sum_{j=1}^{N-n+1} \bar{p}_j Z^{(1)}_{j,n} \\ &= \sum_{j=1}^{N-n} \bar{p}_j \left(E\left[Z^{(1)}_{j,n+1}\middle| X_n = 0,\, X_{n-1} = x\right] + Z^{(2)}_{j,n+1}\right), \end{aligned} \quad \text{(B.21)}$$

where the second equality follows from (B.16), and the last equality is obtained from (B.19) and (B.20). Observe that

$$\begin{aligned} E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1\setminus i})\middle| X_n > 0,\, X_{n-1} = x\right] &= E\left[E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1\setminus i})\middle| X_n > 0,\, X_n,\, X_{n-1} = x\right]\middle| X_n > 0,\, X_{n-1} = x\right] \\ &= E\left[E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1\setminus i})\middle| X_n\right]\middle| X_n > 0,\, X_{n-1} = x\right] \end{aligned} \quad \text{(B.22)}$$

$$= \frac{1}{\alpha} \sum_{j=1}^{N-n+1} \bar{p}_j E\left[Z^{(1)}_{j,n}\middle| X_n > 0,\, X_{n-1} = x\right],$$

where the first equality follows from the tower property of conditional expectation, the second one holds since X_{n-1} becomes redundant once you have the actual value of X_n, and finally, the third equality in (B.22)

is derived from (B.16). Combining (B.18), (B.21), and (B.22) yields

$$\begin{aligned} E\left[R^{\pi^*}(\mathcal{W}_{n-1\setminus i}^{\pi^*})\middle|X_{n-1}=x\right] &= \sum_{j=1}^{N-n}\bar{p}_j\left((1-\alpha)\left(E\left[Z_{j,n+1}^{(1)}\middle|X_{n+1}>0,\,X_n=0\right]+Z_{j,n+1}^{(2)}\right)+\right. \\ &\quad \left. E\left[Z_{j,n}^{(1)}\middle|X_n>0,\,X_{n-1}=x\right]\right) \\ &\quad +\bar{p}_{N-n+1}E\left[Z_{N-n+1,n}^{(1)}\middle|X_n>0,\,X_{n-1}=x\right] \\ &= \sum_{j=1}^{N-n+1}\bar{p}_j\left(E\left[Z_{j,n}^{(1)}\middle|X_n>0,\,X_{n-1}=x\right]+Z_{j,n}^{(2)}\right), \end{aligned} \tag{B.23}$$

where the last equality in (B.23) follows from the definition of $\left\{Z_{j,n}^{(2)}\right\}$ in (B.9) and (B.11). From Hardy's theorem, (B.23), and (B.17), the optimal policy is to assign X_{n-1} to the ith best remaining worker if it falls within the ith highest interval defined by

$$\left\{\left(E\left[Z_{m,n}^{(1)}\middle|X_n>0,\,X_{n-1}\right]+Z_{m,n}^{(2)}\right),\ \text{for all } m=1,2,\cdots,N-n+1\right\}.$$

This result follows since the elements of sets

$$\{\bar{p}_k,\ \text{for all } k=1,2,\cdots,N-n+1\}$$

and

$$\left\{\left(E\left[Z_{m,n}^{(1)}\middle|X_n>0,\,X_{n-1}\right]+Z_{m,n}^{(2)}\right),\ \text{for all } m=1,2,\cdots,N-n+1\right\}$$

are ordered from largest to smallest, and after X_{n-1} is assigned to one of the available workers, the jth highest element of the first set is matched with that of the second set as dictated by the optimal policy (see (B.23)). The assignment of the remaining $N-n+1$ tasks in (B.23) is in accordance with Hardy's theorem, regardless of which worker is assigned to X_{n-1}. Hence, the only way to make all the assignments (including X_{n-1}) consistent with Hardy's theorem, to achieve the maximum sum, is as follows: Match X_{n-1} with the ith smallest-valued worker provided that X_{n-1} is greater than or equal to the $(i-1)$th smallest $\left(E\left[Z_{m,n}^{(1)}\middle|X_n>0,\,X_{n-1}\right]+Z_{m,n}^{(2)}\right)$. In this way, the largest elements of

$$\left\{p_{\pi^*(k,n-1)},\ \text{for all } k=1,2,\cdots,N-n+2\right\}$$

and

$$\left\{X_{n-1},\left(E\left[Z_{m,n}^{(1)}\middle|X_n>0,\,X_{n-1}\right]+Z_{m,n}^{(2)}\right),\forall m=1,2,\cdots,N-n+1\right\}$$

are paired, the next largest are paired, and so forth until the smallest are paired.
Recall that $\left\{\frac{1}{\alpha}Z^{(1)}_{i,n-1},\ \text{for all } i=1,2,\cdots,N-n+2\right\}$ is the set of ordered values of

$$\left\{X_{n-1},\left(E\left[Z^{(1)}_{m,n}\big|X_n>0,\,X_{n-1}\right]+Z^{(2)}_{m,n}\right),\ \text{for all } m=1,2,\cdots,N-n+1\right\},$$

by the definitions given in (B.7)-(B.13). Therefore, the maximum sum obtained by Hardy's theorem and under the above-mentioned optimal policy is

$$E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{n-1})\big|X_{n-1}\right]=\frac{1}{\alpha}\sum_{i=1}^{N-n+2}p_{\pi^*(i,n-1)}Z^{(1)}_{i,n-1}.$$

□

Note that the optimal expected total reward obtained from the SSAP with N tasks and N workers described in the statement of Theorem 22 is given by

$$E\left[R^{\pi^*}(\mathcal{W}^{\pi^*}_{1})\big|X_{1}\right]=\frac{1}{\alpha}\sum_{i=1}^{N}p_{\pi^*(i,1)}Z^{(1)}_{i,1},$$

where $\frac{1}{\alpha}Z^{(1)}_{i,1}$ is the expected value of the task assigned to the ith best worker, $i=1,2,\cdots,N$.

Lemma 19 coincides with the logic behind the Hardy's theorem as it implies that the ith breakpoint upon the arrival of X_n (i.e., $E\left[Z^{(1)}_{i,n+1}\big|X_{n+1}>0,X_n\right]+Z^{(2)}_{i,n+1}$) actually equals the expected value of the task that will be assigned to the ith best remaining worker after the assignment of X_n.

Lemma 19. *For any $1\le n\le N-1$ and $1\le i\le N-n$,*

$$E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_n\right]=E\left[Z^{(1)}_{i,n+1}\big|X_{n+1}>0,X_n\right]+Z^{(2)}_{i,n+1}.$$

Proof. By conditioning on whether $X_{n+1}=0$ or $X_{n+1}>0$, it follows that

$$E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_n\right]=E\left[Z^{(1)}_{i,n+1}\big|X_{n+1}>0,X_n\right]+(1-\alpha)E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right].$$

It only remains to verify that $(1-\alpha)E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right]=Z^{(2)}_{i,n+1}$. To this end, fix an arbitrary $1\le n\le N-1$, and assume that $1\le i\le N-n-1$. First, observe that

$$E\left[Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right]=E\left[Z^{(1)}_{i,n+1}\big|X_{n+1}=0\right],$$

since once X_{n+1} is given, the value of X_n becomes redundant due to the definition of $Z^{(1)}_{i,n+1}$ (see (B.10))

and the fact that the value of each task depends only on the value of the preceding task. Therefore,

$$\begin{aligned}(1-\alpha)E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right] &= (1-\alpha)E\left[E\left[Z^{(1)}_{i,n+2}\big|X_{n+2}>0,X_{n+1}\right]+\right.\\ &\quad \left.Z^{(2)}_{i,n+2}\big|X_{n+1}=0\right]\\ &= (1-\alpha)\left(E\left[E\left[Z^{(1)}_{i,n+2}\big|X_{n+2}>0,X_{n+1}\right]\big|X_{n+1}=0\right]+\right.\\ &\quad \left.Z^{(2)}_{i,n+2}\right)\\ &= (1-\alpha)\left(E\left[Z^{(1)}_{i,n+2}\big|X_{n+2}>0,X_{n+1}=0\right]+Z^{(2)}_{i,n+2}\right)\\ &= Z^{(2)}_{i,n+1},\end{aligned}$$

where the first equality follows from (B.10) and Lemma 17, and the second equality is a direct result of the definition of $Z^{(1)}_{i,n+2}$ (see (B.11)) and the way the dependency between task values is modeled.
Now, assume that $i=N-n$, and note that by (B.8),

$$E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right]=E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0\right]=0,$$

which implies that $(1-\alpha)E\left[\frac{1}{\alpha}Z^{(1)}_{i,n+1}\big|X_{n+1}=0,X_n\right]=Z^{(2)}_{i,n+1}$, by (B.9). □

The *pmf* of the number of arriving tasks N has been considered to have a finite support so far. This assumption is now relaxed, where it is assumed that an infinite positive sequence of arriving tasks with

$$E\left[\sup_n X_n\right]<+\infty,\quad \text{and}\quad \lim_{n\to+\infty}X_n=0, \tag{B.24}$$

should be assigned successively to available workers with success rates $p_1\geq p_2\geq\cdots\geq 0$ such that $\sum_{i=1}^{+\infty}p_i<+\infty$. These assumptions ensure that the expected total reward is finite and well-defined for any policy. The remaining assumptions in the model are the same as before; specifically, a task arrives with probability α during each time period, independently of other time periods. Also, the value of the current task depends on the value of its preceding task.

It is proven that the optimal policy for the infinite-horizon problem is analogous in form to that presented in Theorem 22 for the case with finite number of tasks. More precisely, upon the arrival of the nth task, a set of breakpoints is computed which determines the worker that X_n must be assigned to. It is shown that these breakpoints are the limits, as $N\to+\infty$, of the breakpoints defined for the finite case in Theorem 22.

To this end, some notation and lemmas are needed. Assume that the total number of tasks N is finite, let

$$Y_{m,n}^N := \sup_{\{T_1,T_2,\cdots,T_m\}\in\mathcal{A}_{m,n}} E\left[\sum_{i=1}^{m} X_{T_i}\big|X_n\right], \tag{B.25}$$

for each $1 \le n \le N$ and $m = 1, 2, \cdots, N-n+1$ where $\mathcal{A}_{m,n} := \{\{t_1, t_2, \cdots, t_m\} \mid n \le t_1 < t_2 < \cdots < t_m \le N,\ t_i \in \mathbb{N}\}$. (B.25) basically requires choosing m tasks arriving at time period n or later in order to maximize their expected sum; hence, conditioning on whether X_n should be chosen or not results in

$$Y_{m,n}^N = \max\{X_n + E\left[Y_{m-1,n+1}^N\big|X_n\right], E\left[Y_{m,n+1}^N\big|X_n\right]\},$$

similar to equation (2.1) in [17]. Note that for arbitrarily fixed n and m, $Y_{m,n}^N$ is non-decreasing as N increases. Also, by (B.24)

$$Y_{m,n}^N \le mE\left[\sup_{i\ge n} X_i\big|X_n\right] < +\infty,$$

for any N; therefore,

$$Y_{m,n} := \lim_{N\to+\infty} Y_{m,n}^N \le mE\left[\sup_{i\ge n} X_i\big|X_n\right] < +\infty.$$

To distinguish between problems with distinct number of arriving tasks, a superscript N is added to the breakpoint representations, and hence, $\left\{Z_{m,n}^{(1),N}\right\}$ and $\left\{Z_{m,n}^{(2),N}\right\}$ correspond to a problem with N tasks and N workers. Lemma 20 establishes the relationship between $\left\{Y_{m,n}^N\right\}$ and $\left\{Z_{m,n}^{(1),N}\right\}$. Based on this relationship, it is inferred that the breakpoints defined for the finite case have limits as $N \to +\infty$. This result is then used in Theorem 23 to characterize the optimal policy.

Lemma 20. *For any $1 \le n \le N$ and $1 \le m \le N-n+1$,*

$$Y_{m,n}^N = \frac{1}{\alpha}\sum_{i=1}^{m} Z_{i,n}^{(1),N}. \tag{B.26}$$

Proof. The proof proceeds by backward induction on n and by forward induction on m for each n, and in some parts adopts ideas similar to those of the lemma presented in Section 2 of [17]. Note that

$$Y_{1,N}^N = X_N = \frac{1}{\alpha} Z_{1,N}^{(1),N},$$

which implies that (B.26) is satisfied for $n = N$ and $m = 1$. Now, assume that (B.26) holds for all the pairs (m, n) where $n > d$ and $1 \le m \le N-n+1$ and also for the pairs (m, n) such that $n = d$ and $m \le r$.

Observe that

$$\begin{aligned} Y^N_{r+1,d} &= \max\left\{X_d + E\left[Y^N_{r,d+1}\middle|X_d\right], E\left[Y^N_{r+1,d+1}\middle|X_d\right]\right\} \\ &= \max\left\{X_d + E\left[\frac{1}{\alpha}\sum_{i=1}^{r} Z^{(1),N}_{i,d+1}\middle|X_d\right], E\left[\frac{1}{\alpha}\sum_{i=1}^{r+1} Z^{(1),N}_{i,d+1}\middle|X_d\right]\right\} \\ &= E\left[\frac{1}{\alpha}\sum_{i=1}^{r} Z^{(1),N}_{i,d+1}\middle|X_d\right] + \max\left\{X_d, E\left[\frac{1}{\alpha}Z^{(1),N}_{r+1,d+1}\middle|X_d\right]\right\}, \end{aligned}$$

and note that in order to prove the claim, one needs to show that

$$Y^N_{r+1,d} = \frac{1}{\alpha}\sum_{i=1}^{r+1} Z^{(1),N}_{i,d}. \tag{B.27}$$

To this end, first consider the case where $X_d > 0$, and recall that $\left\{\frac{1}{\alpha}Z^{(1),N}_{i,d},\ \text{for all } i = 1, 2, \cdots, r+1\right\}$ is the set of $r+1$ of largest elements in

$$\left\{X_d, \left(E\left[Z^{(1),N}_{j,d+1}|X_{d+1} > 0, X_d\right] + Z^{(2),N}_{j,d+1}\right),\ \text{for all } j = 1, 2, \cdots, N-d\right\}, \tag{B.28}$$

where (B.28) is equal to the set

$$\left\{X_d, E\left[\frac{1}{\alpha}Z^{(1),N}_{j,d+1}\middle|X_d\right],\ \text{for all } j = 1, 2, \cdots, N-d\right\},$$

by Lemma 19. In addition, $Z^{(1),N}_{j,d+1}$ is a non-increasing function of j, which implies that

$$\left\{\max\left\{X_d, E\left[\frac{1}{\alpha}Z^{(1),N}_{r+1,d+1}\middle|X_d\right]\right\}, E\left[\frac{1}{\alpha}Z^{(1),N}_{j,d+1}\middle|X_d\right],\ \text{for all } j = 1, 2, \cdots, r\right\} \tag{B.29}$$

is the set of $r+1$ of largest elements in (B.28). Therefore, (B.29) and

$$\left\{\frac{1}{\alpha}Z^{(1),N}_{i,d},\ \text{for all } i = 1, 2, \cdots, r+1\right\}$$

are the same sets, which indicates that the sum of their elements is equal; equivalently,

$$r\,\frac{1}{\alpha}\sum_{i=1}^{r+1} Z^{(1),N}_{i,d} = E\left[\frac{1}{\alpha}\sum_{i=1}^{r} Z^{(1),N}_{i,d+1}\middle|X_d\right] + \max\left\{X_d, E\left[\frac{1}{\alpha}Z^{(1),N}_{r+1,d+1}\middle|X_d\right]\right\},$$

and (B.27) is proven for the case where $X_d > 0$. Now, assume that $X_d = 0$, and obtain

$$\begin{aligned}
Y^N_{r+1,d} &= E\left[\frac{1}{\alpha}\sum_{i=1}^{r} Z^{(1),N}_{i,d+1}\big|X_d = 0\right] + \max\left\{0, E\left[\frac{1}{\alpha}Z^{(1),N}_{r+1,d+1}\big|X_d = 0\right]\right\} \\
&= E\left[\frac{1}{\alpha}\sum_{i=1}^{r+1} Z^{(1),N}_{i,d+1}\big|X_d = 0\right] \\
&= \sum_{i=1}^{r+1} E\left[\frac{1}{\alpha}Z^{(1),N}_{i,d+1}\big|X_d = 0\right] \\
&= \sum_{i=1}^{r+1}\left(E\left[Z^{(1),N}_{i,d+1}\big|X_{d+1} > 0, X_d = 0\right] + Z^{(2),N}_{i,d+1}\right) \\
&= \frac{1}{\alpha}\sum_{i=1}^{r+1} Z^{(1),N}_{i,d},
\end{aligned}$$

where the second, fourth, and fifth equalities follow from Lemma 18, Lemma 19, and (B.20), respectively. □

Now that the relationship between $Y^N_{m,n}$ and $Z^{(1),N}_{m,n}$ is established in Lemma 20, the limit of breakpoints defined in (B.7)-(B.13) can be studied. By Lemma 20, $Z^{(1),N}_{m,n} = \alpha\left(Y^N_{m,n} - Y^N_{m-1,n}\right)$, which implies that

$$\bar{Z}^{(1)}_{m,n} := \lim_{N\to+\infty} Z^{(1),N}_{m,n} = \alpha\left(Y_{m,n} - Y_{m-1,n}\right),$$

for $m \geq 1$. Also, set $\bar{Z}^{(1)}_{0,n} := +\infty$. To study the limiting behavior of $Z^{(2),N}_{m,n}$ as $N \to +\infty$, recall that

$$Z^{(2),N}_{m,n} = E\left[\frac{1}{\alpha}Z^{(1),N}_{m,n}\big|X_{n-1}\right] - E\left[Z^{(1),N}_{m,n}\big|X_n > 0, X_{n-1}\right],$$

by Lemma 19. In addition, for each fixed m and n,

$$\left|\frac{1}{\alpha}Z^{(1),N}_{m,n}\right| \leq \left|Y^N_{m,n}\right| + \left|Y^N_{m-1,n}\right| \leq (2m-1)E\left[\sup_{i\geq n} X_i\big|X_n\right],$$

and hence, it follows by the dominated convergence theorem that $\bar{Z}^{(2)}_{m,n} := \lim_{N\to+\infty} Z^{(2),N}_{m,n}$ is well-defined.

For arbitrarily fixed n and $m \geq 1$, values of $\left\{Z^{(1),N}_{m,n}\right\}$ and $\left\{Z^{(2),N}_{m,n}\right\}$ are computed from (B.10) and (B.11) when N gets sufficiently large (specifically, when $N \geq n+m$). Therefore, the dominated convergence theorem results in the following limits for the breakpoints:

$$\begin{aligned}
\bar{Z}^{(1)}_{m,n} &= \alpha\left(X_n \vee \left(E[\bar{Z}^{(1)}_{m,n+1} \mid X_{n+1} > 0,\, X_n] + \bar{Z}^{(2)}_{m,n+1}\right)\right) \wedge \\
&\qquad \left(E[\bar{Z}^{(1)}_{m-1,n+1} \mid X_{n+1} > 0,\, X_n] + \bar{Z}^{(2)}_{m-1,n+1}\right), \\
\bar{Z}^{(2)}_{m,n} &= (1-\alpha)\left(E[\bar{Z}^{(1)}_{m,n+1} \mid X_{n+1} > 0,\, X_n = 0] + \bar{Z}^{(2)}_{m,n+1}\right).
\end{aligned} \tag{B.30}$$

Based on the breakpoint limits established in (B.30), define a policy $\bar{\pi}$ which is of the same form as the optimal policy for the finite case (see Theorem 22) and assigns the nth arriving task to the ith best remaining worker provided that X_n falls within the ith highest interval defined by the breakpoints

$$\left\{E[\bar{Z}^{(1)}_{m,n+1} \mid X_{n+1} > 0,\, X_n] + \bar{Z}^{(2)}_{m,n+1},\ \text{for all } m = 1, 2, \cdots\right\}.$$

Furthermore, let

$$R^{\pi}_n := \sup_{\phi \in \pi|_n} E\left[\sum_{i=1}^{+\infty} p_{\phi_i} X_i \big| X_n, X_{n-1}, \cdots, X_1\right],$$

where $\pi\,|_n$ is the set of all admissible policies which coincide with π before time period n. Observe that R^{π}_n is the optimal conditional expected total reward, given all the task values up to time n, when policy π is applied before time n. By this definition, $\pi\,|_1$ and R^{π}_1 denote the set of all admissible policies and the optimal total expected reward, respectively. Now, define

$$F^{\pi}_n := \sum_{r=1}^{n-1} p_{\pi_r} X_r + \frac{1}{\alpha} \sum_{i=1}^{+\infty} p_{\pi(i,n)} \bar{Z}^{(1)}_{i,n},$$

for each arbitrary policy π. An important property of F^{π}_n, which is used in finding the optimal assignment policy, is presented in Lemma 21. The proof of this lemma is built upon the final part of the proof in Section 3 of [17] in which an optimal policy is characterized for a SSAP with dependent task values where the number of tasks is deterministic and known apriori.

Lemma 21. *For any $\pi \in \pi\,|_1$, $\{F^{\pi}_n : n \geq 1\}$ is a uniformly integrable supermartingale where*

$$F^{\pi}_{\infty} := \lim_{n \to +\infty} F^{\pi}_n = \sum_{r=1}^{+\infty} p_{\pi_r} X_r,$$

and

$$E\left[F^{\pi}_{\infty} \big| X_n, X_{n-1}, \cdots, X_1\right] \leq F^{\pi}_n.$$

Proof. The proof is omitted by analogy to that applied in Section 3 of [17]. □

Theorem 23. *Consider an infinite-horizon SSAP where a sequence of positive tasks satisfying*

$$E\left[\sup_n X_n\right] < +\infty, \quad \textit{and} \quad \lim_{n \to +\infty} X_n = 0,$$

should be assigned to available workers with success rates $p_1 \geq p_2 \geq \cdots \geq 0$ such that $\sum_{i=1}^{+\infty} p_i < +\infty$. A task arrives with probability α in each time period, independently of other time periods; furthermore, values

of any two successive tasks are dependent on each other as defined in (B.2)-(B.5)*. If the assignment of the sequentially arriving tasks to available workers is performed under the policy* $\bar{\pi}$*, then the optimal expected total reward is obtained, and its value is given by*

$$F_1^{\bar{\pi}} = E\left[\sum_{r=1}^{+\infty} p_{\bar{\pi}_r} X_r \big| X_1\right] = \frac{1}{\alpha}\sum_{i=1}^{+\infty} p_{\bar{\pi}(i,1)} \bar{Z}_{i,1}^{(1)}.$$

Proof. The proof technique is similar to that applied in Section 3 of [17], and hence, it is excluded from this dissertation. □

This section presents an optimal policy for SSAP, where a task arrives with probability α during each period (resulting in a Binomial *pmf* for the number of tasks to arrive) and the values of any two consecutive tasks are dependent on each other. An extension of this problem is considered in the next section where the number of arriving tasks is unknown until after the final arrival and is allowed to follow any arbitrary probability distribution.

B.4 Random Number of Tasks

This section generalizes the problem discussed in the previous section to a SSAP where the number of arriving tasks is unknown until after the final arrival and follows an arbitrary probability distribution. Moreover, a task arrives with probability α during each time period, independently of other time periods. Hence, conditional on the total number of tasks, the number of tasks that actually arrive follows a Binomial distribution while the total number of tasks itself follows an arbitrary *pmf*. As before, it is assumed that the values of any two successive tasks are dependent on each other.

Suppose that there are initially N workers available to perform the sequentially arriving tasks where the number of tasks is a random variable with *pmf* f_n. The main challenge in this problem is the randomness in the *number* of arriving tasks. To address this challenge, fix the number of tasks at N_{max}, the largest value that the number of tasks can assume (i.e., $\sum_{n=0}^{N_{max}} f_n = 1$). Given N_{max}, if the number of workers N is less than N_{max}, then one can add $N_{max} - N$ phantom workers having success rates of 0 associated with them. On the other hand, if $N > N_{max}$, the $N - N_{max}$ workers with the smallest values can be dropped so that the number of workers equals N_{max}. Another modification necessary to tackle this problem is to augment the state space of task values by adding a new state 0_{ab}. Once the process reaches this state, no more tasks arrive, which implies that the number of tasks has reached its maximum and the sequential assignment terminates. The value of a task in this state is assumed to be zero. Note that the state 0 is distinct from 0_{ab} in that if the state at time period j is 0, then there is a positive probability α that $X_{j+1} > 0$ (i.e., a task arrives during the next period) while when the state is 0_{ab}, then $X_k = 0_{ab}$ for all $k > j$. Similar to

the argument originally used in [22], the probability distribution governing the dependency between any two consecutive task values is therefore given by:

$$\begin{aligned} P_\alpha^f\{X_{n+1}=0 \mid X_n=k\} &= P_\alpha^f\{X_{n+1}=0 \mid X_n=k, N>n\}\, P_\alpha^f\{N>n \mid X_n=k\} \\ &= P_\alpha\{X_{n+1}=0 \mid X_n=k\}\, P_\alpha^f\{N>n \mid N\geq n\} \\ &= g_\alpha(0 \mid k)\frac{\sum_{i=n+1}^{N_{max}} f_i}{\sum_{i=n}^{N_{max}} f_i}, \end{aligned} \tag{B.31}$$

$$\begin{aligned} P_\alpha^f\{X_{n+1}\in\mathcal{B} \mid X_n=k\} &= P_\alpha^f\{X_{n+1}\in\mathcal{B} \mid X_n=k, N>n\}\, P_\alpha^f\{N>n \mid X_n=k\} \\ &= P_\alpha\{X_{n+1}\in\mathcal{B} \mid X_n=k\}\, P_\alpha^f\{N>n \mid N\geq n\} \\ &= g_\alpha(\mathcal{B} \mid k)\frac{\sum_{i=n+1}^{N_{max}} f_i}{\sum_{i=n}^{N_{max}} f_i}, \end{aligned} \tag{B.32}$$

and

$$P_\alpha^f\{X_{n+1}=0_{ab} \mid X_n=k\} = P_\alpha^f\{N=n \mid N\geq n\} = \frac{f_n}{\sum_{i=n}^{N_{max}} f_i}, \tag{B.33}$$

for $k \in \mathcal{S}\cup\{0\}$ and $\mathcal{B}\subseteq\mathcal{S}$ where P_α^f implies that the conditional probability is computed under the assumption that f is the *pmf* for the total number of tasks while a task arrives with probability α during each time period. Moreover,

$$P_\alpha^f\{X_{n+1}=0_{ab} \mid X_n=0_{ab}\}=1, \tag{B.34}$$

which implies the termination of the task-arrival stream. Once the probability distribution is updated (as in (B.31)-(B.34)) to incorporate randomness in the number of tasks, the interval breakpoints can be computed by (B.7)-(B.13). Then the theorems presented in Section 3 are applied to determine the optimal assignment of the arriving tasks to available workers.

B.5 Numerical Results

As mentioned before, a special case of the model introduced in Section B.3 is a SSAP in which the dependency between task values are governed by a Markov chain. The following example illustrates a SSAP with a finite number of tasks (i.e., $N<+\infty$) where tasks arrive with probability α during each time period. The values of any two consecutive tasks are dependent on each other where the dependecy structure is given by the following transition probability matrix:

$$P\{X_1=y\}=r_y \quad \text{with} \quad y\in\mathcal{S}\cup\{0\} \quad \text{such that} \quad \sum_{y\in\mathcal{S}\cup\{0\}} r_y=1 \quad \text{and} \quad r_0=1-\alpha, \tag{B.35}$$

$$P\{X_{n+1} = 0 \mid X_n = y\} = q_{y0} = 1 - \alpha \quad \text{with} \quad y \in \mathcal{S} \cup \{0\}, \tag{B.36}$$

$$P\{X_{n+1} = l \mid X_n = y\} = q_{yl} \quad \text{with} \quad y \in \mathcal{S} \cup \{0\} \quad \text{and} \quad l \in \mathcal{S} \quad \text{such that} \quad \sum_{l \in \mathcal{S}} q_{yl} = \alpha, \tag{B.37}$$

for $n = 1, 2, \ldots, N-1$ where $\mathcal{S} \subseteq (0, +\infty)$ is a finite set. Therefore, (B.35), (B.36) and (B.37) define a discrete-time Markov chain with a finite state space $\mathcal{S} \cup \{0\}$ where the state at time j represents the task value at that time.

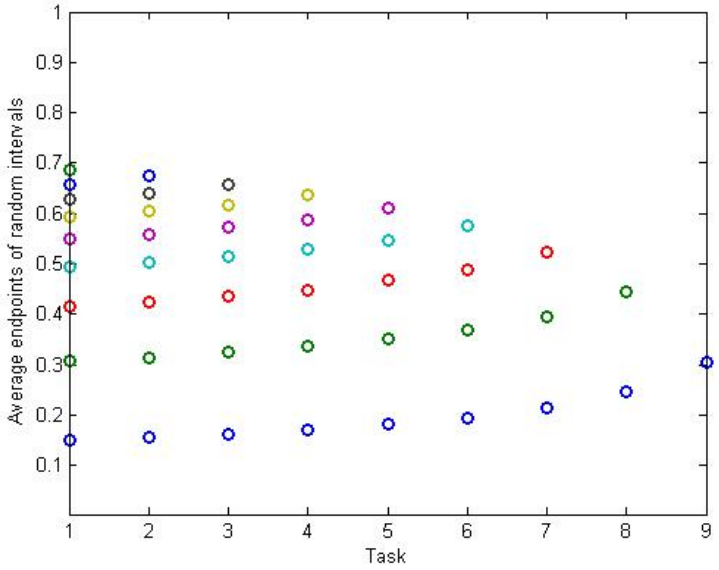

Figure B.1: Average of endpoints of real line's random intervals

In this example, task values, success rates of the workers, q_{yl}, and r_y (with $y \in \mathcal{S} \cup \{0\}$ and $l \in \mathcal{S}$) follow $U[0, 1]$ distributions, independently of one another. The number of tasks N and the number of states of the Markov chain are set equal to 10 with $\alpha = 0.4$. The endpoints of the random intervals which the real line is partitioned into (under the optimal assignment policy) are computed using simulations of 1000 replications, and an average is captured. Figure B.1 depicts the average of the endpoints of these intervals, which the real line is partitioned into upon the arrival of each task. For this example, where $N = 10$, the endpoints of random intervals are computed for the first nine arriving tasks, with the last task being assigned to the last remaining worker. Specifically, when the jth task arrives at time period j, $N - j$ breakpoints are illustrated in Figure B.1 along the Y-axis. Depending on the value of the arriving task, these breakpoints are used in choosing the best worker to be assigned to this task.

Furthermore, the performance of our model (taking dependency and random number of tasks into consideration) is compared to that of the classic SSAP model introduced by [12]. To this end, 100000 replications

of the above-mentioned problem are generated under a common Markov chain governing the dependencies, and each replication is solved using the optimal policy presented previously in this work. For each problem (i.e., each sample path), a similar problem is solved by applying the classic SSAP policy proposed by [12], assuming that the simulated task values are IID. Specifically, the stationary distribution of the Markov chain is taken to be the underlying *pmf* of the simulated task values, which are assumed to be IID. Note that this is only an approximation, and the task values (in each sample path) are actually dependent on each other through the Markov chain transitions. This stationary distribution is used to calculate the threshold values proposed by [12], and task assignments are performed based on these breakpoints. Then, the average expected reward (over all the 100000 replications) from these two sets of problems are compared to each other. The results indicate a %40 increase in the expected total reward under the optimal policy presented here, compared to the classic SSAP (approximate) policy. The next section discusses concluding remarks and future directions of research.

B.6 Conclusion

This part of the dissertation analyzes the SSAP under the assumption that tasks arrive with a certain probability in each time period and values of any two successive tasks are dependent on each other. An optimal assignment policy is provided in order to sequentially assign workers to arriving tasks so as to maximize the expected total reward. The complexity of the proposed algorithm is the same as the complexity of the original algorithm introduced by [12].

In addition, an extension of the problem is studied where the number of arriving tasks is unknown until after the final arrival and follows a finite-support generic *pmf*. The problem is then generalized to the case where the underlying *pmf* of the number of tasks has infinite support, while the values of any two consecutive tasks are dependent on each other. It is proven that the optimal policy to maximize the expected total reward in the infinite-horizon model is obtained by taking the limit of the policy for the finite-horizon case.

In this thesis, it is assumed that the distribution of the random variable associated with task values is known apriori. Further research is required to address the SSAP in which task values follow a distribution with an unknown parameter. Also, the proposed model assumes that at each time period the value of a task depends only on the value of the preceding task. Finding an optimal policy for a model with random number of arriving tasks where a more general dependency structure exists between task values is yet another problem to be studied.

References

[1] J. Ahn and J. J. Kim. Action-timing problem with sequential Bayesian belief revision process. *European Journal of Operational Research*, 105:118–129, 1998.

[2] S. C. Albright. A markov-decision-chain approach to a stochastic assignment problem. *Operations Research*, 22(1):61–64, Jan. - Feb. 1974.

[3] S. C. Albright. Optimal sequential assignment with random arrival times. *Management Science*, 21(1):60–67, 1974.

[4] S. C. Albright. A markov chain version of the secretary problem. *Naval Research Logistics Quarterly*, 23(1):153–159, March 1976.

[5] S. C. Albright. A Bayesian approach to a generalized house selling problem. *Management Science*, 24(4):432–440, December 1977.

[6] S. C. Albright. A Bayesian approach to a generalized house selling problem. *Management Science*, 24:432–440, 1977.

[7] S. C. Albright and C. Derman. Assymptotic optimal policies for stochastic sequential assignment problem. *Management Science*, 19(1):46–51, 1972.

[8] G. Baharian and S.H. Jacobson. Limiting behavior of the target-dependent stochastic sequential assignment problem. *Naval Research Logistics*, 60:321–330, 2013.

[9] G. Baharian and S.H. Jacobson. Stochastic sequential assignment problem with threshold criteria. *Probability in the Engineering and Informational Sciences*, 27:277296, 2013.

[10] T. Biswas and J. McHardy. Asking price and price discounts: the strategy of selling an asset under price uncertainty. *Theory and Decision*, 62(3):281–301, May 2007.

[11] K. Boda, J. A. Filar, Y. Lin, and L. Spanjers. Stochastic target hitting time and the problem of early retirement. *IEEE Transactions on Automatic Control*, 49(3):409–419, March 2004.

[12] C. Derman, G. J. Lieberman, and S. M. Ross. A sequential stochastic assignment problem. *Management Science*, 18(7):349–355, March 1972.

[13] C. Derman, G. J. Lieberman, and S. M. Ross. A stochastic sequential allocation model. *Operations Research*, 23(6):1120–1130, 1975.

[14] C. Derman, G. J. Lieberman, and S. M. Ross. A stochastic sequential allocation model. *Operations Research*, 23(6):1120–1130, 1975.

[15] A. Gershkov and B. Moldovanu. Efficient sequential assignment with incomplete information. *Games and Economic Behavior*, 68:144–154, 2010.

[16] G. Hardy, J. Littlewood, and G. Polya. *Inequalities*. Cambridge University Press, Cambridge, 2nd edition, 1959.

[17] D. P. Kennedy. Optimal sequential assignment. *Mathematics of Operations Research*, 11(4):619–626, 1986.

[18] A. Khatibi, G. Baharian, and S.H. Jacobson. Doubly stochastic sequential assignment problem. *working paper*.

[19] L. A. McLay, S.H. Jacobson, and A. G. Nikolaev. A sequential stochastic passenger screening problem for aviation security. *IIE Transactions*, 41(6):575–591, 2009.

[20] S.P. Meyn and R.L. Tweedie. *Markov chains and stochastic stability*. Springer-Verlag, London, 1993.

[21] T. Nakai. A sequential assignment problem in a partially observable markov chain. *Mathematics of Operations Research*, 11:230–240, 1986.

[22] A. G. Nikolaev and S.H. Jacobson. Stochastic sequential decision-making with a random number of jobs. *Operations Research*, 58:1023–1027, 2010.

[23] A. G. Nikolaev, S.H. Jacobson, and L. A. McLay. A sequential stochastic seceurity system design problem for aviation security. *Transportation Science*, 41(2):182–194, May 2007.

[24] Y. Ohtsubo. Value iteration methods in risk minimizing stopping problems. *Journal of Computational and Applied Mathematics*, 152:427–439, 2003.

[25] Y. Ohtsubo. Optimal threshold probability in undiscounted markov decision processes with a target set. *Applied Mathematics and Computation*, 149:519–532, 2004.

[26] Y. Ohtsubo and K. Toyonaga. Optimal policy for minimizing risk models in markov decision processes. *J. Math. Anal. Appl.*, 271:66–81, 2002.

[27] Y. Ohtsubo and K. Toyonaga. Equivalence classes for optimizing risk models in markov decision processes. *Math. Meth. Oper. Res.*, 60:239–250, 2004.

[28] R. L. Righter. The stochastic sequential assignment problem with random deadlines. *Probabability in the Engineering and Informational Sciences*, 1:189–202, 1987.

[29] M. Sakaguchi and Y. Ohtsubo. Optimal threshold probability and expectation in semi-markov decision processes. *Applied Mathematics and Computation*, 216:2947–2958, 2010.

[30] J. E. Smith and K. F. McCardle. Structural properties of stochastic dynamic programs. *Operations Research*, 50(5):796–809, 2002.

[31] X. Su and S. A. Zenios. Patient choice in kidney allocation: A sequential stochastic assignment model. *Operations Reaserch*, 53(3):443–455, 2005.

[32] D. J. White. Minimizing a threshold probability in discounted markov decision processes. *J. Math. Anal. Appl.*, 173:634–646, 1993.

[33] C. Wu and Y. Lin. Minimizing risk models in markov decision processes with policies depending on target values. *J. Math. Anal. Appl.*, 231:47–67, 1999.